Mémoire

Auteur : Laisney, Clément
Promoteur(s) : Sluse, Dominique; Delchambre, Ludovic
Faculté : Faculté des Sciences
Diplôme : Master en sciences spatiales, à finalité approfondie
Année académique : 2022-2023
URI/URL : http://hdl.handle.net/2268.2/17440

University of Liège

Departement of Astrophysics, Geophysics and Oceanography
Astrophysics and Geophysics Institut

Master in Space Sciences, Research focus

Innovative techniques to find strongly lensed systems

Laisney Clément

Promotor Sluse Dominique
Cosmologics and Astrophysics Origins
University of Liège

Co-Promotor Delchambre Ludovic
Group of AstroPhysics and High-Energies
University of Liège

June 2023

Abstract

A galaxy-galaxy lens is a phenomenon in which the light of a background distant galaxy is deflected in the vicinity of a massive foreground galaxy. The occurrence of this phenomenon is very rare. The advent of big-data surveys is an opportunity to detect gravitational lenses only if the proper tools are built. This work aims to build such a tool by testing a set of innovative techniques using parametric and non-parametric models to identify the presence of lensed galaxies in a dataset. The dataset used in this work is a simulated dataset combining true galaxy images with lens simulations also based on true galaxies. Based on this simulated dataset, we use simple machine learning algorithms like Support Vector Machine (SVM), Random Forest (RF), or Multi-Layer Perceptron (MLP). This simple method is an asset for the comprehension of the classification process compared to Convolutional Neural Networks (CNN) that are commonly used. In this exploratory work, we found that 91.8% of the simulated lenses are reported and classified with a precision of 95.6%. This work reports thus promising results using MLP that are lower but rather close to the performances of a CNN. This work can be improved to reach comparable or even better results than today's state-of-the-art algorithms by studying residual image resulting from the substraction of a light profile to the original image.

Acknowledgement

Even if this work is personal work, it involves people who I would like to thank. First of all, I would like to thank Dominique Sluse and Ludovic Delchambre who were my two promoters. They were real support in this work, with good advice. I really appreciated the rigor that they succeeded in transmitting to me and which I sometimes miss. The next people I want to thank are my parents, my family, my girlfriend, and my friends who were also a real support in my studies. I especially grateful to my parents who allowed me to have this incredible academic path without ever constraining my choices. Finally, I would like to thank all the students, teachers, researchers, and administrators I have met and who have enriched my personal journey, my thoughts, and my experience.

Contents

1 Introduction

A gravitational lens, also known as a cosmic mirage, is a distribution of mass able to bend the light coming from a distant source. This phenomenon is similar to a light beam bent through a lens by refraction. This is why we call it gravitational lensing.

Already in 1704, Isaac Newton speculated that "[...] Bodies act upon Light at a distance, and by their action bend its Rays; and [...] this action [is] strongest at the least distance." [1]. Later John Mitchell proposed to Henry Cavendish, a method to measure the mass of stars by detecting a reduction of the speed of light affected by gravity [2]. In those letters, Mitchell suggested that a massive enough body could stop the light: a black hole. This pushed Cavendish to calculate the Newtonian light deflection. Unfortunately, he never published his manuscript dated around 1784 [3]. Johann Georg von Soldner published the same result in 1801 [4] assuming the light is a corpuscle. Finally, Einstein calculated the same value thanks to the equivalence principle only in 1911 [5] and corrected it by twice the value in 1915 [6] in the frame of General Relativity.

The first observation of light deflection was performed in 1919 by Arthur Eddington and Frank Watson Dyson by observing a change in the position of stars near the sun during the solar eclipse of May 29 [7]. Later in 1937, after the new discovery of Galaxies, Fritz Zwicky speculated that those massive objects could act as both source and lens with a larger effect much likely to be observed [8]. It was necessary to wait until 1979 to observe the first gravitational lens. Dennis Walsh, Bob Carswell, and Ray Weysmann observed two identical QuasiStellar Objects (QSO) using Kitt Peak National Observatory. The difficulties in describing them as two distinct objects were highlighted in their paper [9], along with the discussion of the hint suggesting the observation of two images of the same object formed through gravitational lensing. SBS 0957+561 was renamed "Twin QSO".

1.1 Scientific context

Lenses used to be serendipitously found, but nowadays we are looking for gravitational lenses in large amounts thanks to large survey programs. One can cite an early example of systematic research of gravitational lenses: the Cosmic Lens All-Sky Survey (CLASS). Twenty-two lensed systems were found using the Very Large Array (VLA) radio telescope [10].

We generally subdivide lensing into 3 categories: Strong, Weak, and Micro-lensing. Strong lensing is the case where distortion of the background source is clearly identified or multiple lensed images are detected. For weak lensing, distortion is much smaller, and statistical studies are needed in order to find a distortion of about a few percent. The microlensing does not show any distortion but a variation of the background source light over time. The lensing object in a microlensing case may be a star while strong and weak lenses are typically galaxies or even galaxy clusters. Microlensing is often used to detect exoplanets [11]. In this work, I will focus on strongly lensed systems.

Studying gravitational lenses has a huge scientific interest. As F. Zwicky already mentioned in his paper in 1937 [8], they constitute a good source to test general relativity, they enable us to observe very distant galaxies and they allow us to determine masses of galaxies. In fact, it has many more applications like the determination of cosmological parameters (Ω_0: density parameter, λ_0: cosmological constant, and H_0: the Hubble constant). But the two main fields of interest are the study of dark matter and dark energy [12] [13], and modern cosmology [14] by better-determining cosmological distance scales, large-scale matter distribution, mass distribution in galaxy clusters, physics of quasars and galaxy structure.

With the rise of Large surveys and their ability to store a large amount of data, we are facing Big Data problems. As an example, the Euclid Consortium website[1] asserts that during the 6-year-long mission, Euclid will collect more than 500,000 visible and Near-Infrared images. In addition to that, ground-based telescopes will cover the same sky as Euclid in 4 different filters which represent a total of 7 different filters. This forms several millions of images that weigth 30 PetaByte of data. About 10 billion galaxies will be imaged during the mission. Other surveys like DES[2] (Dark Energy Survey) or LSST[3] deal with the same amount of data. About 700 million sources in the last DES data release and 2 million images (60 PetaByte)

[1] https://www.euclid-ec.org/?page_id=2625
[2] https://www.darkenergysurvey.org/
[3] https://www.lsst.org/

for LSST. Processing this tremendous amount of data requires detecting lensed systems autonomously. The estimated occurrence of galaxy-galaxy strong lensing is 10^{-5}[15][16]. In most optimistic forecasts, DES, LSST, and Euclid can discover up to 2300, 120000, and 280000 lenses respectively [15]. The low occurrence of the phenomenon forces us to build observational and computational strategies to maximize the chances of detection. The common strategy adopted today is the rise of lens candidates from survey data to be followed up later by dedicated ground-based telescopes. Because observation time is precious we clearly see the need for a robust autonomous model rising the least possible false positive cases.

1.2 Physics of Gravitational lenses

This section will focus on the mathematical demonstration of the gravitational lensing phenomenon. To do so, my development is largely inspired by Jean-François Claeskens and Jean Surdej's review [17] and Pierre Magain's lecture [18].

1.2.1 The Refraction analogy

As said earlier, J. von Soldner calculated the deviation angle of light by a spherical mass M thanks to Newton's theory of gravitation. This deviation angle $\alpha = \frac{2GM}{c^2 b}$ (where G is the gravitational constant, c the speed of light, and b the impact parameter.) was corrected later by Einstein thanks to general relativity. In this part, we will study gravitational lenses in the frame of General relativity.

Gravitational mirages are analogs of atmospheric mirages. This phenomenon arises when light trajectories are curved (scheme in fig 1.1) which is a consequence of an anisotropic speed of light along the light path. According to refraction laws, light rays are bent in an inhomogeneous medium. This is why we can approach this problem as a refraction problem, with light traveling from vacuum to a material medium of refraction index n_ϕ. The refractory analog situation is summarized in fig 1.2.

In vacuum; the speed of light is c, but in a material medium it becomes $v = \frac{c}{n_\phi}$. As a consequence of General relativity, spacetime is curved by a gravitational potential ϕ associated with a massive object. In the weak field approximation ($\frac{\phi}{c^2} \ll 1$), the metric is :

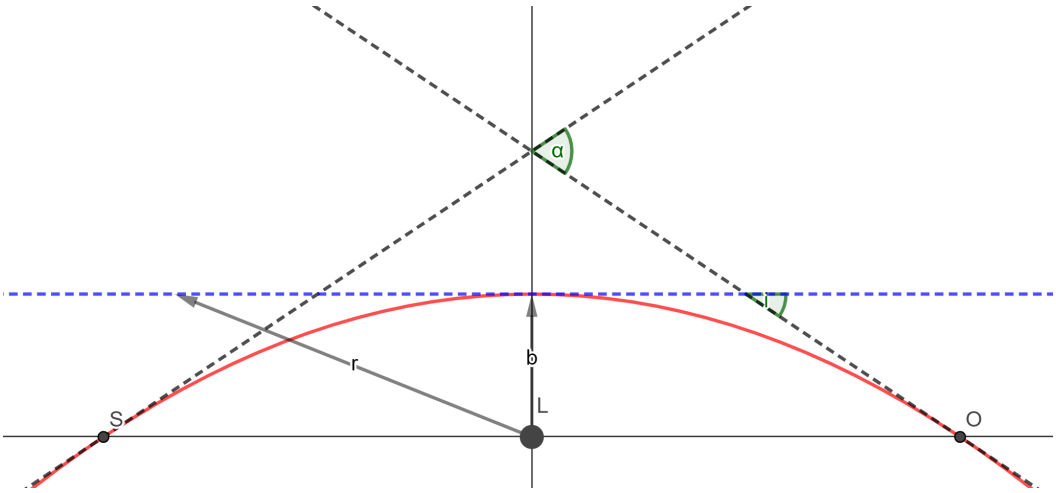

Fig. 1.1.: Deflection of light coming from a distant source (S) in the vicinity of a massive object (Lens: L) seen by the observer (O).

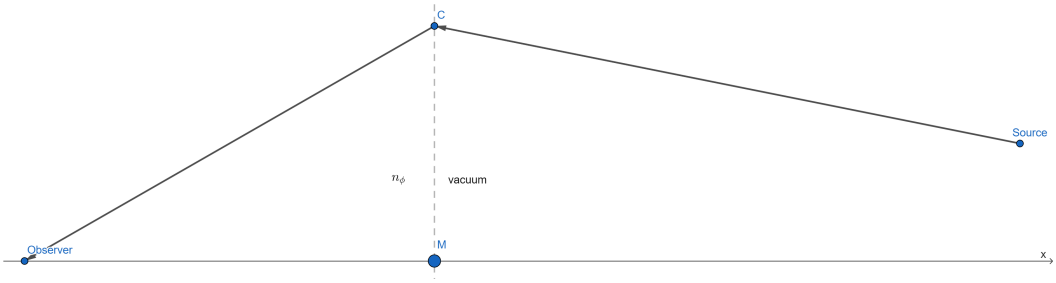

Fig. 1.2.: Analog situation of light traveling in vacuum from a distant source and propagating through a medium of refraction index n_ϕ.

$$ds^2 = -\left(1 + \frac{2\phi}{c^2}\right) c^2 dt^2 + \left(1 - \frac{2\phi}{c^2}\right) (dx^2 + dy^2 + dz^2) \tag{1.1}$$

As the light follows null geodesic, we have that $ds^2 = 0$ such that:

$$\left(1 + \frac{2\phi}{c^2}\right) c^2 dt^2 = \left(1 - \frac{2\phi}{c^2}\right) (dx^2 + dy^2 + dz^2) \tag{1.2}$$

$$dt^2 = \frac{1 - \frac{2\phi}{c^2}}{1 + \frac{2\phi}{c^2}} \frac{(dx^2 + dy^2 + dz^2)}{c^2} \tag{1.3}$$

$$dt^2 \approx \left(1 - \frac{2\phi}{c^2}\right)^2 \frac{(dx^2 + dy^2 + dz^2)}{c^2} \tag{1.4}$$

$$dt \approx \left(1 - \frac{2\phi}{c^2}\right) \frac{\sqrt{dx^2 + dy^2 + dz^2}}{c} \tag{1.5}$$

$$\frac{\sqrt{dx^2 + dy^2 + dz^2}}{dt} \approx \frac{c}{\left(1 - \frac{2\phi}{c^2}\right)} \tag{1.6}$$

equation 1.6 is analog to $v = \frac{c}{n_\phi}$. We identify $n_\phi = 1 - \frac{2\phi}{c^2}$. That being said, let's establish the lens equation and the deflection angle.

1.2.2 The lens equation

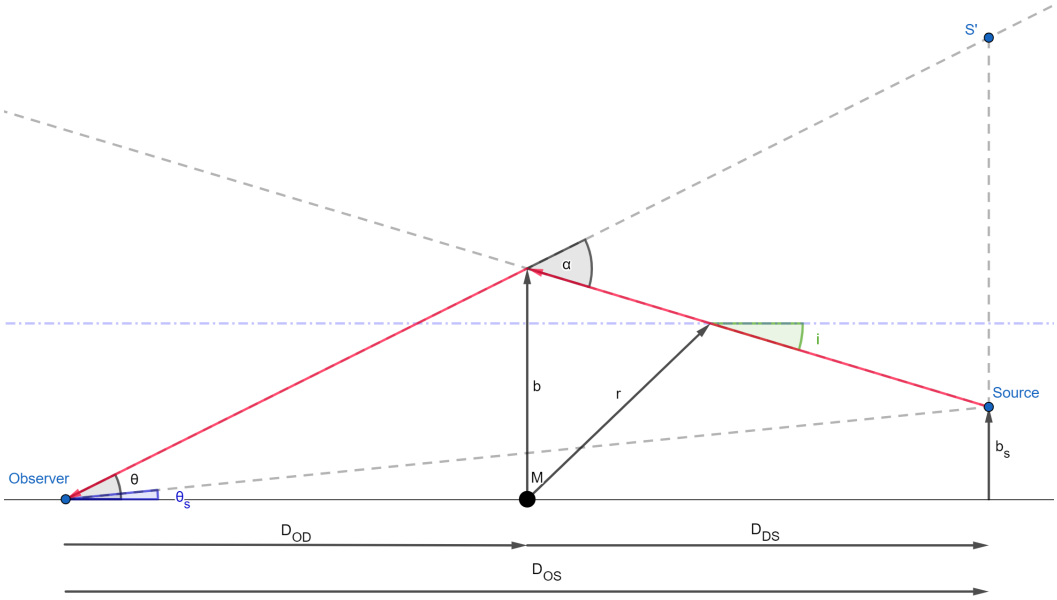

Fig. 1.3.: Scheme of the general situation of a gravitational lens

To solve this problem we need to link the viewing angle θ to the deflection angle α. By geometric considerations thanks to figure 1.3 this relation is given by the following equation :

$$\theta_s = \theta - \frac{D_{DS}}{D_{OS}}\alpha(b) \tag{1.7}$$

$$b = D_{OD}\theta \tag{1.8}$$

Now we have to link the deflection angle α to physical parameters like the mass of the deflector M. In the following, $\frac{di}{dx}$ is the variation of the direction along the x-axis.

α is thus the deflection angle which is the integration of all direction variations along x.

$$\alpha(b) = -\int_{-\infty}^{\infty} \frac{di}{dx} dx = -\int_{-\infty}^{\infty} \frac{1}{n_\phi} \frac{dn_\phi}{db} dx = \frac{2}{c^2} \int_{-\infty}^{\infty} \frac{d\phi}{db} dx \tag{1.9}$$

In this demonstration, we will take as a simple example a point mass deflector. Its gravitational potential is :

$$\phi = -\frac{GM}{r} = -\frac{GM}{\sqrt{b^2 + x^2}} \tag{1.10}$$

The deflection angle α becomes:

$$\alpha(b) = \frac{4GM}{c^2 b} \tag{1.11}$$

We effectively find twice the value obtained with the Newtonian framework. Now that we find what Einstein predicted, let's generalize. To do so, we use the thin lens approximation. This approximation allows us to describe a deflector by its surface mass density $\Sigma(\overrightarrow{b})$ projected in the deflector plane. The deflection angle is then expressed by :

$$\overrightarrow{\alpha}(\overrightarrow{b}) = \frac{4G}{c^2} \int_S \frac{\Sigma(\overrightarrow{b'})(\overrightarrow{b} - \overrightarrow{b'}) db'_1 db'_2}{|\overrightarrow{b} - \overrightarrow{b'}|^2} \tag{1.12}$$

In the case of a circularly symmetric mass distribution, with $M(b)$ the mass inside the radius b and $b = \| \overrightarrow{b} \|$:

$$\overrightarrow{\alpha}(\overrightarrow{b}) = \frac{4GM(b)}{c^2 b^2} \overrightarrow{b} \tag{1.13}$$

1.2.3 Einstein Ring

Let's now assume the following circularly symmetric lens mass distribution with M_0, the mass inside a radius b_0:

$$M(b) = M_0 \left(\frac{b}{b_0} \right)^\beta \tag{1.14}$$

$\beta = 0$ correspond to a point mass distribution, $\beta = 1$ is the singular isothermal sphere distribution and $\beta = 2$ yield the uniform distribution of matter.

Now that we have the general case, let's study a particular case which is the Einstein ring solution. This case corresponds to the simple case when the observer, the

deflector, and the source are aligned ($\theta_s = 0$). The Einstein ring is thus defined by its angular size θ_E. From equations 1.7, 1.13 and 1.14 we have that:

$$0 = \theta_E - \frac{D_{DS}}{D_{OS}} \frac{4GM(b)}{c^2 b} = \theta_E - \frac{D_{DS}}{D_{OS}} \frac{4G}{c^2 b} M_0 \left(\frac{b}{b_0}\right)^{\beta} \tag{1.15}$$

$$0 = \theta_E - \frac{D_{DS}}{D_{OS}} \frac{4GM_0}{c^2 b_0^{\beta}} \left(D_{OD}\theta_E\right)^{\beta-1} = \theta_E \left(1 - \frac{D_{DS}}{D_{OS}} \frac{4GM_0}{c^2 b_0^{\beta}} D_{OD}^{\beta-1} \theta_E^{\beta-2}\right) \tag{1.16}$$

$\theta_E = 0$ is a solution but not relevant.

$$\frac{D_{DS} D_{OD}^{\beta-1}}{D_{OS}} \frac{4GM_0}{c^2 b_0^{\beta}} \theta_E^{\beta-2} = 1 \tag{1.17}$$

$$\theta_E = \left(\frac{4GM_0}{c^2 b_0^{\beta}} \frac{D_{DS}}{D_{OD}^{1-\beta} D_{OS}}\right)^{\frac{1}{2-\beta}} \tag{1.18}$$

with M_E the mass inside the radius $b_E = D_{OD}\theta_E$ we get the expression of the Einstein ring angular size:

$$\theta_E = \sqrt{\frac{4GM_E}{c^2} \frac{D_{DS}}{D_{OD} D_{OS}}} \tag{1.19}$$

1.2.4 Images positions

By tacking a point mass deflector ($M(b) = M_0$):

$$\theta_s = \theta - \frac{D_{DS}}{D_{OS}} \frac{4GM_0}{c^2 b} = \theta - \frac{D_{DS}}{D_{OS}} \frac{4GM_0}{c^2 D_{OD} \theta} \tag{1.20}$$

$$\theta\theta_s = \theta^2 - \frac{D_{DS}}{D_{OS} D_{OD}} \frac{4GM_0}{c^2} \tag{1.21}$$

with $M_E = M_0$

$$\theta^2 - \theta_s \theta - \theta_E^2 = 0 \tag{1.22}$$

we are in the presence of a second-order polynomial with a determinant $\delta = \theta_s^2 + 4\theta_E^2 > 0$ and its two solutions:

$$\theta_{1,2} = \frac{1}{2}\left(\theta_s \pm \sqrt{\theta_s^2 + 4\theta_E^2}\right) \tag{1.23}$$

Let's now study two interesting properties. By considering a small misalignment ϵ between the source, the lens, and the observer; at first order, the angular separation of the image $\Delta\theta$ is:

$$\theta_1 = \frac{1}{2}\left(\epsilon + \sqrt{\epsilon^2 + 4\theta_E^2}\right) = \frac{1}{2}\epsilon + \frac{1}{2}\sqrt{4\theta_E^2} = \frac{1}{2}\epsilon + \theta_E \tag{1.24}$$

$$\theta_2 = \frac{1}{2}\epsilon - \theta_E \tag{1.25}$$

$$\Delta\theta = \theta_1 - \theta_2 = 2\theta_E \tag{1.26}$$

This means that the angular separation between 2 images increases with the mass of the deflector or increases when the deflector is closer to the observer.

The second property is about the mean surface density within θ_E which remains constant. We define this quantity as the critical surface mass density:

$$\bar{\Sigma}(\theta_E) \equiv \frac{M_E}{\pi(D_{OD}\theta_E)^2} = \frac{c^2 D_{OS}}{4\pi G D_{OD} D_{DS}} \equiv \Sigma_{crit} \tag{1.27}$$

This property implies that a finite massive object is a gravitational lens that can produce multiple images if its central mass density $\Sigma > \Sigma_{crit}$

1.2.5 Magnification

Another property of gravitational lenses is the magnification. This quantity μ is given by the ratio between the surface brightness of the image and the source. In the following, we will stay in a 1D case but the magnification can be generalized using Jacobian given by this definition:

$$\mu = \frac{d\Omega_{image}}{d\Omega_{source}} = \left| det\left(\frac{\partial\theta_s}{\partial\theta}\right)\right|^{-1} \tag{1.28}$$

Assuming a circular symmetry of the lens as in the figure 1.4: $\mu = \frac{\theta}{\theta_s}\frac{d\theta}{d\theta_s}$

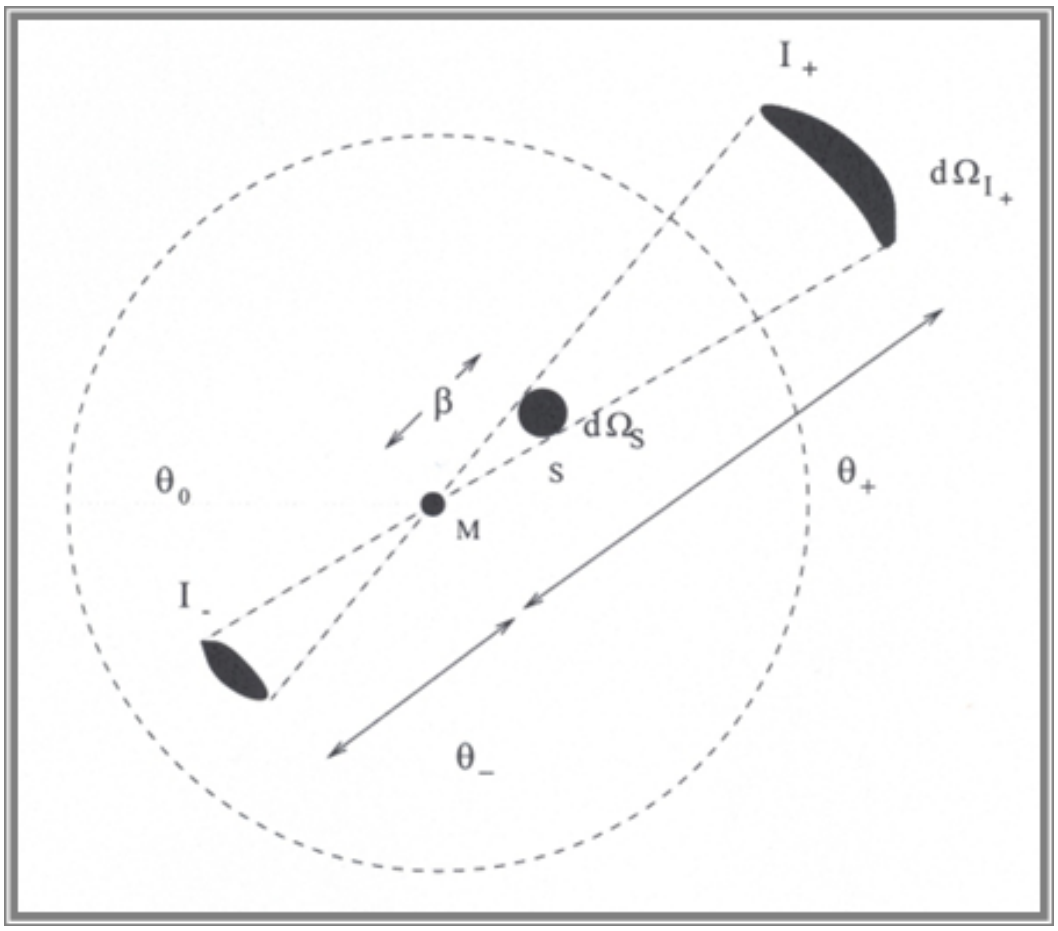

Fig. 1.4.: Illustration of a gravitational lens with magnification for a point source from [18]. β correspond to the angle between deflector and source which is θ_s in the previous demonstration. θ_- and θ_+ are respectively the demagnified and magnified images positions. The dotted circle corresponds to the Einstein ring position.

by considering $u = \frac{\theta_s}{\theta_E}$ the lens-image separation in units of Einstein ring radius, we get:

$$\mu_\pm = \frac{u^2+2}{2u\sqrt{u^2+4}} \pm \frac{1}{2} \tag{1.29}$$

The $+$ solution is always magnified while the $-$ solution can be magnified or demagnified depending on the value of u. If the source is inside the Einstein radius, $\mu > 1.34$.

Finally, the total magnification gives:

$$\mu = \mu_+ + \mu_- = \frac{u^2+2}{u\sqrt{u^2+4}} > 1 \tag{1.30}$$

For a point-like source, if $u \to 0$, $\mu \to \infty$. If $u \to \infty$, $\mu \to 1$. In the case of too-small image separation, the magnification can still be measured. This is used when the mass of the deflector is too small to dissociate the image from the lens. But it is also used when the image and the lens separation cannot be resolved and this technique is called micro-lensing.

1.2.6 Properties of the targets

To better understand the scales of the targets, the distances at stake, and the physical properties of the object of interest, I will here give some orders of magnitudes that can help understand the different challenges.

The main targets we are looking for are galaxies or clusters of galaxies but in our frame of work, we will focus on galaxy-galaxy strong lensing. The total mass of a galaxies can span from 10^7 to 10^{12} solar masses [19] [20]. The distance scale is often expressed as a redshift value. To have an idea, $z = 1$ is approximately 20 Billion light years (this is a comoving distance). Typical distances for galaxies are of the order of $z \equiv 1$.

To have an idea of the angular separation between 2 images let's take a deflector mass of 10^{12} solar masses and a distance of $z \equiv 0.5$. For the source let's consider a quasar at a distance of $z \equiv 2$. Thanks to relation 1.19 and 1.26, we have in arcseconds:

$$\Delta\theta = 2\theta_e \approx 1.8\sqrt{\frac{M_E}{10^{12}M_{sun}}} \tag{1.31}$$

which in our case gives $\Delta\theta \approx 1.8$". This represents approximately $\frac{1}{20}$ of the angular size of Jupiter. Galaxies are very faint objects in the sky and can span a large range of absolute magnitude, the Andromeda galaxy has an apparent magnitude of 3.5 while the faintest known galaxy has an apparent magnitude of 28 mag.

In other words, our targets are very massive, far-away objects such that the angular size is about an arcsecond with faint magnitudes. Combined with the occurrence of 10^{-5}, it perfectly illustrates the prowess to find this type of targets

1.3 Convolutional Neural Network to the rescue

With the era of large-scale surveys, Astrophysics has to deal with a huge amount of data and faces similar problems as the Big Data industry. As the amount of data grows, Machine learning and Deep learning field improved at the same time allowing the industry to manage the processing of those data.

The main type of data in Astrophysics is images and this is especially the case for the study of gravitational lenses. Automatically detecting gravitational lenses requires methods able to detect localized features. Convolutional Neural Networks (CNN) are well-suited for this task. Convolution layers are able to detect local correlations with features learned from the data[21].

CNNs have proven their efficiencies in many fields as computer vision and self-driven cars [22] but also in research fields like medicine [23] and biology [24]. This is why so many attempts at auto-detection of gravitational lenses use CNNs. Obviously, there are other methods[25] investigated but CNNs are widely at the core of the research of strong lensed systems.[16][26][27][28][29].

1.4 A new approach

This work aims to explore innovative techniques to automate the detection of gravitational lenses. This time, CNNs will be discarded, but parametric and non-parametric models will be explored and combined with the power of other machine-learning methods. We will deeply explore one family of methods but we will later discuss different ones that came to our mind.

One of the shortcomings of the CNN approach results in a loss of interpretability. Indeed, neural networks are often considered black boxes even if we can intuitively understand what is going on inside of these black boxes. In this master thesis, I will first make a data reduction by using models and morphological indexes, and then I will interpret this data reduction thanks to machine learning algorithms that aim to classify the images from this data. The model I mostly use among many others is the Sersic model [30] which has the advantage of being more understandable. We will also use common non-parametric and morphological indexes like Asymmetry, Concentration, Gini[31], and many more[32][33]. I will try to interpret these Sersic parameters plus morphological indexes to see the variations of these parameters in images containing a lens.

With the help of machine learning techniques, I will try to classify them in an autonomous way. During this work, I needed a controlled framework that is not provided with real data. This control is important in the sense that I want to clearly know what is the influence of a lens on our parameters. To do so, I will need a lot of different cases to identify the values that our indexes can span. Unfortunately, too few real lensed galaxies cases are available nowadays and it can have a significant impact on our parameters since it can confine in a smaller range of values than it could span in reality. The next reason is that it is impossible to find the same lensed and non-lensed galaxy image in real life. This pair of lensed/unlensed galaxy images are highly relevant to my work because it allows me to check the origin of a subset of parameters and clarify whether it is caused by the presence of the lens or if it originates from features present in the non-lensed galaxy image. My dataset contains both non-lensed and simulated galaxy-galaxy strong gravitational lensing from true galaxy images. It ensures both control and proximity with true data. Section 2.1 will go deeply into the conception of the dataset.

One of the tricks in this work is to fit a Sersic model on a galaxy candidate and detect the presence of a lens thanks to the difference with the non-lensed model. The Sersic profile is mainly defined by 3 parameters: the amplitude I_e, the effective radius R_{eff}, and the Sersic index n:

$$I(R) = I_e \exp\left(-b_n \left[\left(\frac{R}{R_{\text{eff}}}\right)^{\frac{1}{n}} - 1\right]\right) \tag{1.32}$$

From [34], $b_n = 1.9992n + 0.3271$ for $0.5 < n < 10$. We will thus constrain the values of non-lensed Sersic parameters (a "standard" distribution); and in case a gravitational lens is present, the luminosity in the arms of the galaxy will increase and these parameters will move away from the "standard" distribution enabling us to detect the lens. In the parameter space, we should observe a separation of point clouds allowing us to discriminate non-lensed from lensed galaxies. By using another data reduction technique, a different parametrization of the light profile (morphological indexes) will be computed and will be processed the same way. To well understand the idea of this method, a comparison of 3 fitted Sersic profiles on 3 non-lensed galaxies and the same lensed galaxies and Sersic profiles is displayed in fig 1.5. Already in this figure, we can notice changes in the amplitude I_e, the Sersic index n, and the effective radius R_{eff} when the lens is present. We will discuss expected trends of this changes in section 3.2. As the Sersic model is a smooth model the variations in the two different cases are not visually striking (even if the amplitude can be multiplied up to 5 in the fig 1.5), however, the azimutal profile is

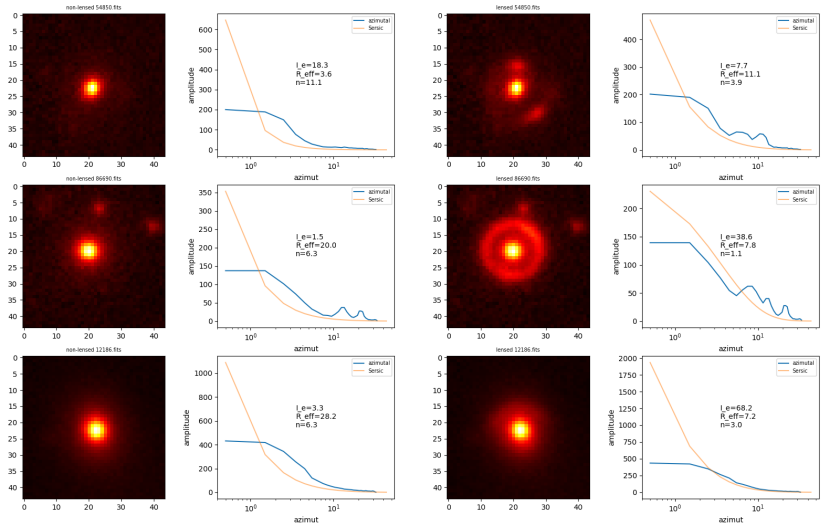

Fig. 1.5.: Comparison between fitted Sersic profiles and azimutal profile in the case of non-lensed galaxies (left) and lensed one (right). Azimutal profile is the maximum of the pixel values on a circle of a given radius from the center of the galaxy. The presence of contaminating source or lens images results in a bump in the azimutal profile while the Sersic model tries to fit to fit it with a smoother model. I_e, R_eff, and n stands for the fitted amplitude, effective radius and Sersic index (respectively).

also scattered for each image to help understand what is going on. The azimutal profile (as I defined it) is the maximum of the pixel values on a circle of a given radius from the center of the galaxy (the azimut). We can observe that the azimutal profile notifies the presence of a contaminating galaxy or just the lens image by a bump in the curve. And we see, that the Sersic profile attempt to fit as close as possible to the azimutal profile. For example the presence of multiple bumps in the azimutal profile number `86690` implies a change in the steepness of the Sersic profile.

In this report, we will start by describing data. In a second time, we will focus on the methods used to perform this work. And before a conclusion, we will discuss our results and the other methods we could have used.

2 Data

To test my method, I needed simulated data as discussed in section 1.4. Instead of making self-made simulated data, I worked with a dataset originally made for a CNN-based study. This dataset is the E. Savary, K. Rojas, M. Maus, et al dataset [28]. Its design suits our requirements and can be easily transposed to our framework. This part thus aims to explain the design of the dataset and have a better understanding of the data. The first step is to know how the dataset was collected and designed. The second step is to make a sanity check of the dataset and identify the different possible cases. And the last step is to identify the nature of the signal and the associated statistics.

2.1 Origin of the Data

A part of the images constituting our simulated dataset was acquired as part of the Canada France Imaging Survey (CFIS), which is a component of the Ultraviolet Near Infrared Optical Northern Survey (UNIONS). UNIONS is a Collaboration open to Canadian, French, and Japanese PhD students, astronomers, and members of the Pan-STARRS team. This collaboration aims to provide answers about dark matter, galactic to cluster scaled structures of the Universe, and the assembly of the Milkyway. This collaboration will contribute to ground-based photometry within the Euclid ESA space mission[1].

CFIS mainly uses the Canada-France-Hawaii Telescope (CFHT) to complete a survey of 8000 deg^2 of the northern sky in u photometric band (CFIS-u) and 4800 deg^2 in the r photometric band (CFIS-r). CFHT is a world-class 3.6 m optical/infrared telescope located on top of the Mauna Kea volcano in Hawaii[2]. It is equipped with MegaCam which is a wide-field optical imaging device built from 40 2048x4612 pixels CCD cells. MegaCam has a total of 380 Megapixels for a square Field-of-View (FoV) of 1 deg^2. The resolution of the telescope is 0.187 arcsecond/pixel.[3]

[1] https://www.skysurvey.cc/
[2] https://www.cfht.hawaii.edu/en/about/
[3] https://www.cfht.hawaii.edu/Science/CFIS/

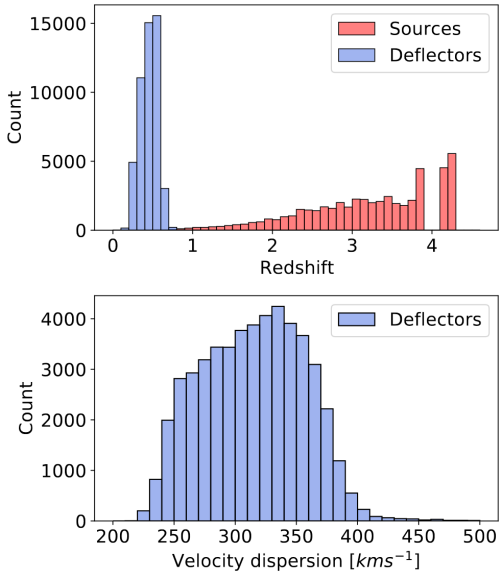

Fig. 2.1.: Redshit distribution of LRG galaxies (blue) and background lensed galaxies (red). The LRG velocity dispersion is important for Einstein radius computation (see 2.1 for more details). From [28]

The dataset we use is simulated images from real data coming from the CFIS Data Release 2 and HST/ACS F814W images. The way it is built is the following: CFIS images are used as deflectors galaxies (fig 2.2) and Hubble Space Telescope (HST) images lie in background sources that are lensed by software (fig 2.3). Images size is a square of 44 pixels which represents an FoV of 8.17". Images from CFIS are exclusively from CFIS-r data and more specifically Luminous Red Galaxies (LRG). These objects are expected to have the largest lensing cross section due to their bright and massive properties. Background galaxies were converted in the r photometric band with HSC ultra-deep stacked images. Images have an FoV of 10" for each side which corresponds to 0.03" per pixel.

Foreground galaxy images (LRG) have SDSS spectra allowing us to get access to velocity dispersion σ_{v} and a redshift z estimate available in the image headers (Image headers store metadata of the image). Background images which are all included in COSMOS2015 [35] and Galaxy Zoo catalog [36] were obtained using public spectroscopic catalogs or estimated from the best photometric redshift from [35]. According to figure 2.1, the foreground LRG galaxies spans the ranges $200 < \sigma_{\mathrm{v}} < 500\ km\ s^{-1}$ and $0.1 < z < 0.7$. It is important to notice that already lensed LRG galaxies are present in the dataset but the effects on the performances of doubled lensed or misclassified images are considered negligible knowing the occurrence of the phenomenon (10^{-5}).

In this work, like in the reference paper, we will perform classification and thus train machine learning classifiers to identify gravitational lenses. To do so we define positive cases and negative cases. Negative cases are non-lensed galaxies randomly drawn from LRG galaxies sample described previously belonging to CFIS-r data. While positive cases are built from the combination of a foreground galaxy (CFIS) and a background-lensed galaxy (HST). The main steps of the design of positive cases follow these main steps:

- Random selection of an LRG
- Assignment of a singular isothermal ellipsoid mass model to the LRG galaxy.
- Random selection of a galaxy from HST sample
- Position of the source is randomly chosen with total magnification $\mu \geq 2$
- Computation of high-resolution image of the lensed source
- Convolution of the lensed source with the CFIS Point Spread Function (PSF)
- Combination of the deflector and the lensed source images

Let's go deeper into details about the different steps and the assumption and choices made during the process. The first assumption is the assignation of a mass profile model. The model used is the Singular Isothermal Ellipsoid (SIE) model; a generalization of the Singular Isothermal Sphere (SIS) model. This model is the simplest distribution of matter in galaxies. The formula ruling this model is given by:

$$\rho_{\mathrm{SIS}}(r) = \frac{\sigma_{\mathrm{v}}^2}{2\pi G r^2} \tag{2.1}$$

The angular size of the Einstein radius can be linked to the velocity dispersion by taking this equation at $r = R_E$ and $R_E = D_{OD}\theta_E$:

$$\rho_{\mathrm{SIS}}(\theta_E) = \frac{\sigma_{\mathrm{v}}^2}{2\pi G D_{OD}^2 \theta_E^2} \tag{2.2}$$

This model can then be generalized to an ellipse and is parameterized by five free parameters: Einstein radius, center coordinates of the lens, ellipticity, and position angle. Since every image was designed to host the deflector center at the center of the image, the center coordinates are a fixed value. The ellipticity and the position angle (PA) were derived from the second moment of the light profile of the LRG. In this model, it is assumed that the ellipticity and the PA derived from the light distribution are the same as the mass distribution profile.

This being done, a source galaxy is randomly selected within the HST sample. with the redshift information of both galaxies and the velocity dispersion, the Einstein radius is computed thanks to the previous relation linking θ_E to σ_v. The value of Einstein's radius (see equation 1.18) is imposed so that the angular size falls in the interval $0.8" < \theta_E < 3.0"$. The lower limit is chosen to prevent lens-galaxy blending while the upper limit is set to fit into the image frame. If the angular size of the Einstein radius is outside the frame of the image, another source is chosen. After 100 iterations the dispersion velocity of the deflector is increased by 50%. If the conditions are not met after this process, the deflector is discarded from the dataset. As explained in the paper [28], this velocity dispersion boost involves a few objects with small velocity dispersion and it is not expected to introduce a significant morphological bias.

The next systematic choice made during the building process is the total magnification constraint. E. Savary, K. Rojas, M. Maus, et al decided to impose $\mu \geq 2$ because it coincides with the threshold for a multiple-image lens[37]. Setting a higher limit would have increased the number of full Einstein rings among the simulations and this could lead to an increase of false positives by mistaking ring galaxies.

When all those conditions are met, a high-resolution image of the lensed source is computed and convolved with the CFIS PSF. Here the HST point spread function is neglected since it is sharper than the CFIS PSF. Before applying the CFIS PSF one, it is important to re-sample the PSF to the HST pixel size and then down-sampled to the CFIS pixel size after the convolution. The final step is the addition of the lensed source image to the foreground galaxy image.

2.2 Exploration of the dataset

Our dataset is made of multiple files divided into 5 types of data. We have first `LRG only` images which are the CFIS-r images visible in fig 2.2. The second type of image is `Lensed source only` images which are the lensed images of an HST source galaxies as shown in fig 2.3. The third photometric-based images are the combination of `LRG only` and `Lensed source only` images which gives `Lens simulation` images. Figure 2.4 represent those images which are the combination of images in fig 2.2 and 2.3. The two other types of data are PSF and RMS files. PSF files contain the two-dimensional function associated with the response of a focused imaging system (here the Canada France Hawaii Telescope) to a point source. We only get the PSF from CFHT but as mentioned in section 2.1 we neglected the HST PSF. Finally, RMS

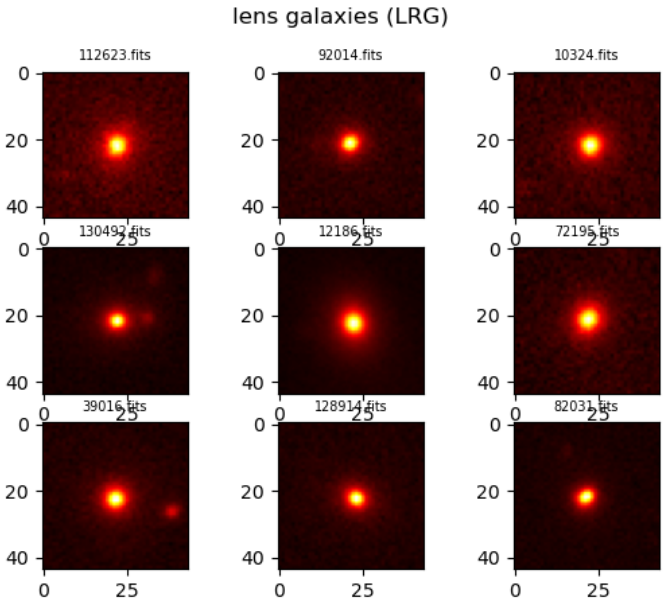

Fig. 2.2.: Sample of 9 images from the CFIS-r dataset. Those images are used as lens galaxies during the design of the dataset. Images are displayed with a linear red colormap.

files correspond to Root Mean Square errors associated with each pixel. RMS files were built from weight maps computed from gain and relative normalization of noise using `SWarp`[4] software. It is important to note that we only get RMS files associated with `LRG only` images. We will see later how to generate RMS files for `Lens simulation` files. A sample of PSF and RMS files can be found in figures 2.5 2.6.

2.3 Noise and signal

Imaging galaxies is not as simple as taking photographs in daily photography. Imaging deep sky objects implies low light intensity where different noises become non-negligible. Those noises arise due to different physical phenomena that we won't emphasize here because they have been corrected already. However, we will focus on the photon noise which persists in data and cannot be canceled with calibration files so it is important to quantify it.

The photon noise is linked to the physical nature of the light and affects the quality of the data. This phenomenon is not perceptible by the human eye because of the retinal

[4] `https://www.astromatic.net/software/swarp/`

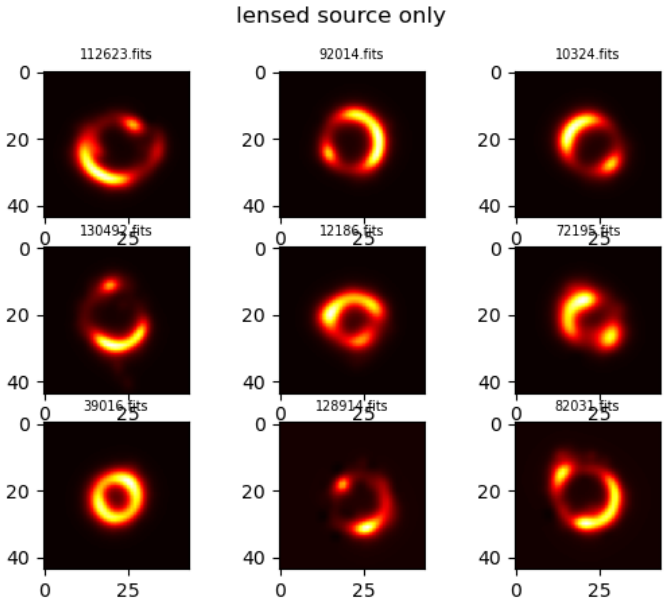

Fig. 2.3.: Sample of 9 simulated lens images from the HST source galaxies dataset. Those images are used as lens features during the design of the dataset. Images are simulated lenses according to lens galaxies displayed in fig 2.2.

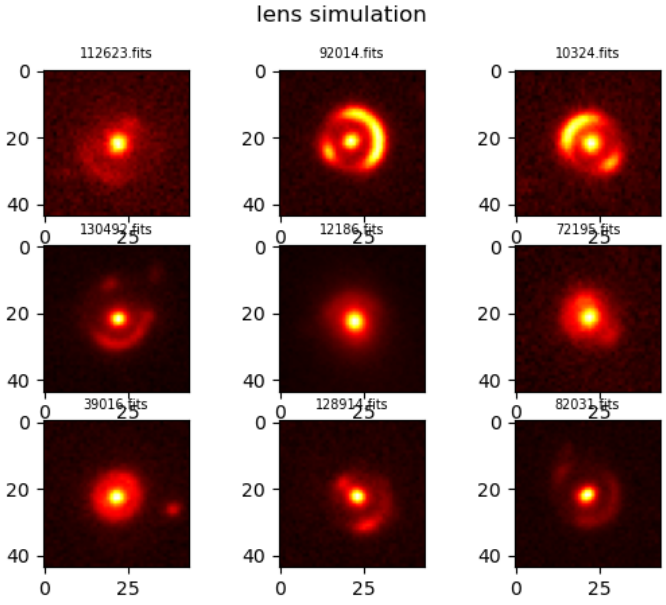

Fig. 2.4.: Sample of 9 simulated images from the CFIS-r and HST dataset. Those images are addition of fig 2.2 and 2.3 images.

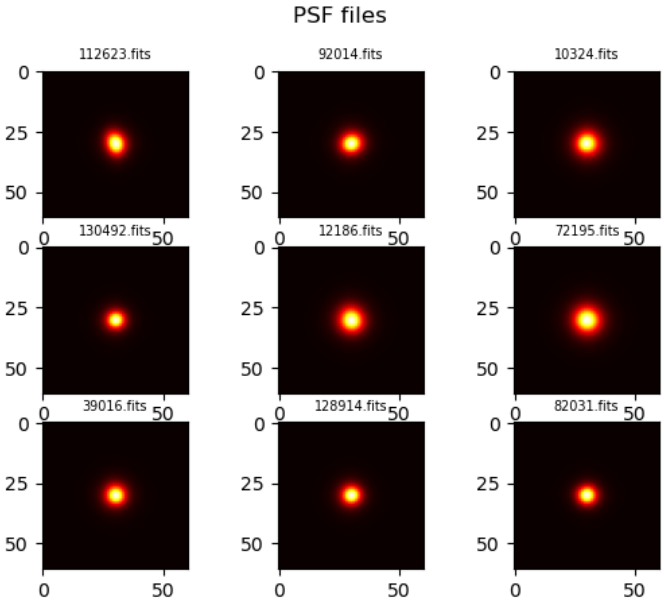

Fig. 2.5.: Sample of 9 PSF images from the Canada France Hawaii Telescope.

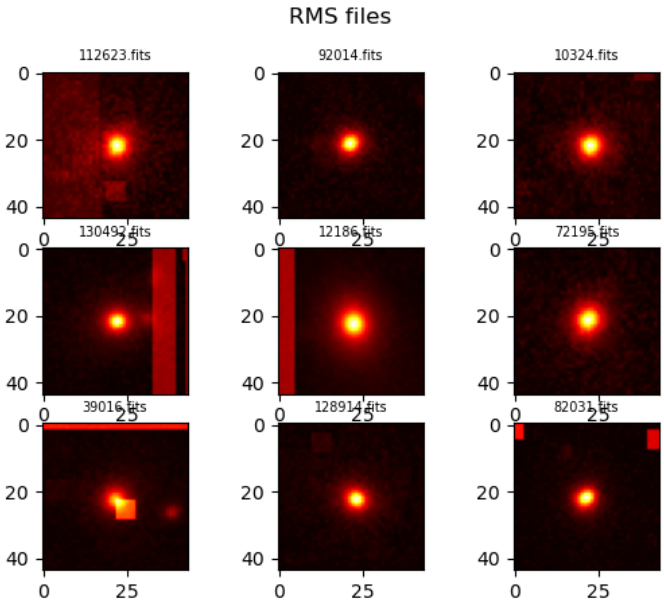

Fig. 2.6.: Sample of 9 RMS images associated with CFIS-r images. Squares on some images result from a mask of satellite tracks, cosmic rays or dead pixels.

persistence that cancels irregularities in the photonic flux. The flux of photons is a random variable that is associated with a counting problem. The statistics associated with a counting phenomenon are modeled by Poisson distribution. We can thus evaluate the noise $\sigma_{photons}$ by the following equation, where N is the number of detected photons:

$$\sigma_{photons} \propto \sqrt{N} \tag{2.3}$$

Pixel intensities are expressed as ADU units which means Analog-to-Digital-Units. One can easily switch from ADUs to the number of electrons thanks to the gain $g = \frac{\text{number of electrons}}{ADU}$. We can thus link the intensity in ADU to the intensity in electrons I_{e-} units with the gain g.

$$I_{ADU} = I_{e-} \times g \tag{2.4}$$

Thanks to equation 2.3, $I_{e-} = \sigma_e^2 = RMS^2$ and we then get the following relation:

$$RMS^2 = \frac{I_{ADU}}{g} \tag{2.5}$$

In our case, we can consider that the image is made of photons from the object of interest and photons coming from the background sky $I_{ADU} = I_{obj,ADU} + I_{sky,ADU}$. In object regions and their neighborhood, the intensity of the sky is very low or even negligible compared to the object intensity and $I_{ADU} \approx I_{obj,ADU}$ And the total Root Mean Square error is :

$$TOTAL\ RMS^2 = \sigma_{obj}^2 + \sigma_{sky}^2 \approx \frac{I_{ADU}}{g} + \sigma_{sky}^2 \tag{2.6}$$

It is important to notice that in regions away from an object, $I_{sky,ADU} \sim 0$ for background substracted images and $TOTAL\ RMS^2 \approx \sigma_{sky}^2$

If `LRG only` images are well background substracted, we expect the relation 2.6 to fit to the data. In order to check that, let's fit this linear relation on plotted pixel values (in ADU) as a function of RMS^2 as it can be found in fig 2.7. Before discussing the results of this graph, let's focus on the general trend and features of the point clouds only. As discussed a few lines earlier, lower values of pixels, imply a dominant background noise and lead to a plateau that is present in fig 2.8. It is common to find profiles as displayed in fig 2.9 with points away from the point cloud with the

computed gain (fit)	real gain	σ^2_{sky} (fit)	σ^2_{sky} (plateau)	σ^2_{sky} (corner method)
33.27	31.58	13.43	13.56	12.12
42.36	42.36	3.84	3.88	3.19
44.79	44.49	6.77	6.81	4.00
41.43	39.04	3.98	4.11	2.72
53.92	53.97	4.12	4.58	9.35
42.12	41.83	4.29	4.35	3.73
39.16	46.55	4.82	4.86	2.78
52.78	52.75	3.83	3.88	3.53
28.93	29.03	5.36	5.48	4.16

Tab. 2.1.: Comparison of values computed for the gain and the background noise. fit values stand for values computed thanks to linear regression, plateau by tacking the mean value of the plateau and corner method by the method described in section 2.3. The real gain is the one available in the image header.

same trend but translated along the y-axis. It is due to one or several masks present in the RMS file (see fig 2.6) increasing error bars in case of a polluting source like satellite tracks, cosmic rays, or even dead pixels.

To enhance the accuracy of the fit of relation 2.6 on the scattered points, I discarded points belonging to the plateau at low pixel values. However, we will also evaluate the background noise by only considering points of this plateau. So, a sample of 10 fits is displayed in fig 2.7, and values of the computed gain and computed σ^2_{sky} with different techniques are available in table 2.1.

Fitted gain values are compared to the gain values available in the image headers. For σ^2_{sky} we compared fitted values with the evaluated mean value of the so-called plateau. I also evaluated it by another method that I call the corner method. The corner method consists in taking some pixel values at the 4 corners of the image where the background is dominant. I took a square kernel of 5 pixels large at each corner so we have a total of 100 pixels. To obtain the background noise, I took the squared standard deviation of those pixels.

Coming back to table 2.1 results, we clearly see that the gain values computed are coherent with the ones found in image headers and we also have the same evaluations for the background noise. It is particularly true between the fit and the plateau method. However, corner methods only consider 100 pixels which is a small sample of the total background sky, and can thus not confidently evaluate σ^2_{sky}. Moreover, there is a non-zero probability that a contaminating neighbor source (galaxy or star) is located in the corner (see an example in images fig 2.2) which should have an effect on this estimation.

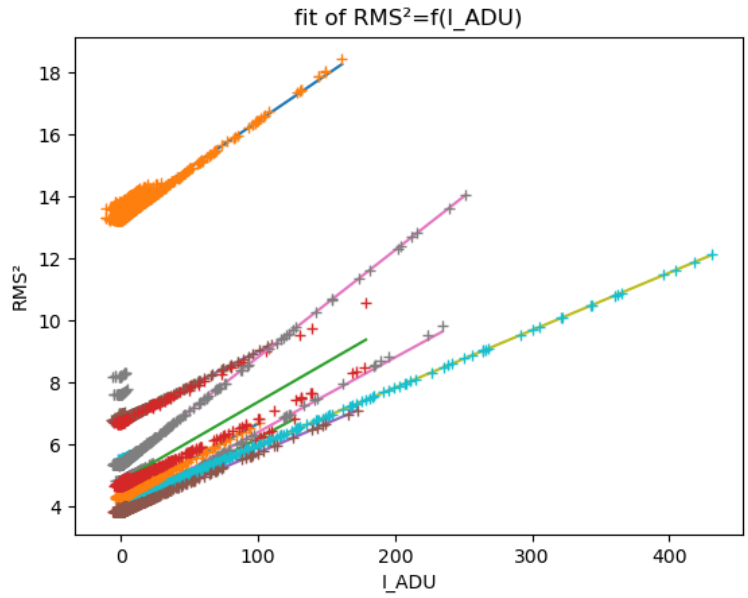

Fig. 2.7.: Linear regression of 9 `LRG only` RMS^2 as a function of pixel intensity in ADU using relation 2.6.

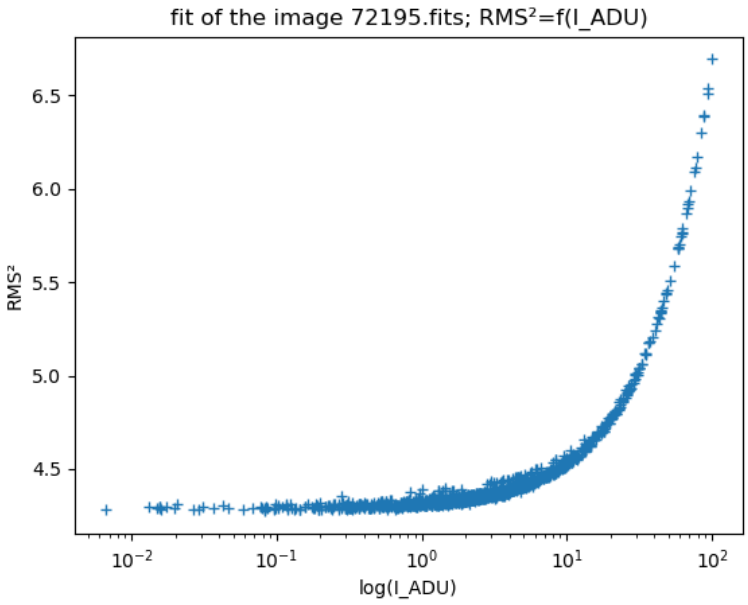

Fig. 2.8.: RMS^2 as a function of pixel intensity in log(ADU) for `LRG only` images

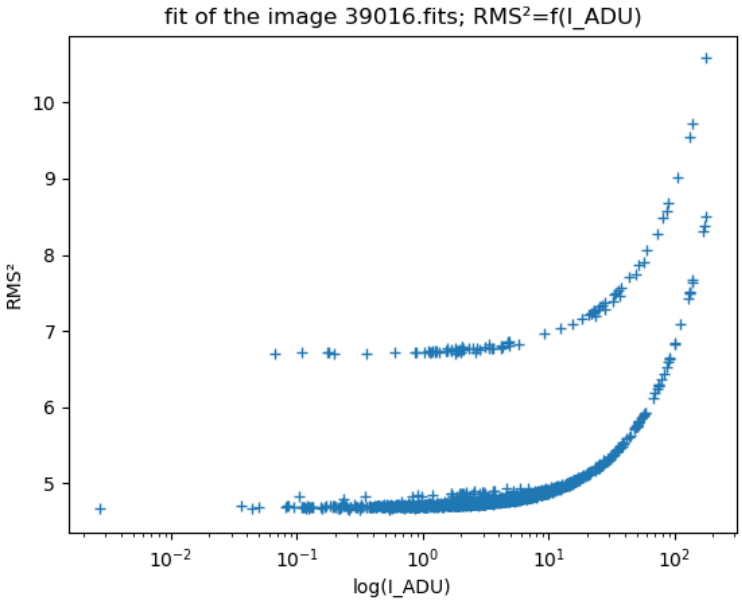

Fig. 2.9.: Pixel value in log(ADU) as a function of RMS^2 for `LRG only` images with polluting source and RMS mask. Here the higher values correspond to the RMS masks with higher RMS values due to an increase of the error. This error is mainly due to the presence of a polluting source which can take the form of a satellite track, a cosmic ray or a dead pixel.

In the following of this work, we will need RMS files for `Simulated images`, but we only have RMS files for `LRG only` images. Now that we know a bit more about the link between the signal and the noise, it is easy to build a new RMS file for `Simulated images`. But before that, I checked that images are a simple addition of `LRG only` and `Lensed source only` images by substracting both files from the simulated one. So based on equation 2.5, I reconstructed the RMS file with the quadratic sum of the already existing RMS file and a computed Mean Square error using the gain value available in the `LRG only` headers. Here is the expression of the new RMS:

$$RMS_{reconstructed} = \sqrt{RMS^2 + \frac{I_{ADU,Lensed\_source\_only}}{g_{LRG}}} \tag{2.7}$$

3 Methodology

In this chapter, I will address the process I built to perform an autonomous classification of lensed galaxies. The first step in the process is the detection of the source of interest in order to fit a Sersic model and compute common non-parametric indexes, as we already mentionned in section 1.4. The next step is the training of a machine learning algorithm able to differentiate non-lensed and lensed galaxies from previously computed parameters. This chapter will first discuss the detection of the main source, next the choice of parameters, and finally the choice of machine learning models.

3.1 Detection of sources: segmentation maps, threshold, and masks.

Source detection is an important step in my work since the quality of computed parameters will influence the performance of the classifier. We will discuss this influence in chapter 4. In order to detect sources I use `photutils` [38] which is an astronomical python package for photometry. This package includes a function that detects sources above a specified detection threshold value in an image: `detect_sources`[1]. This threshold is an absolute value that needs to be computed for each image. The `detect_source` function takes as input the image data, a threshold value that we will explore in the following, and a number of connected pixels that have to be connected and higher than the threshold value to be considered as part of the source. `detect_sources` returns a segmentation map which is a two-dimensional array of the same size as the image. This segmentation map marks the position of pixels part of a detected source by a positive integer associated with a segment while 0 stands for the background. Segmentation files can be found in fig 3.1.

In the documentation of the `detect_sources` function, it is recommended to use a built-in deblending function after the use of `detect_sources`. This deblending

[1] `https://photutils.readthedocs.io/en/stable/api/photutils.segmentation.detect_sources.html`

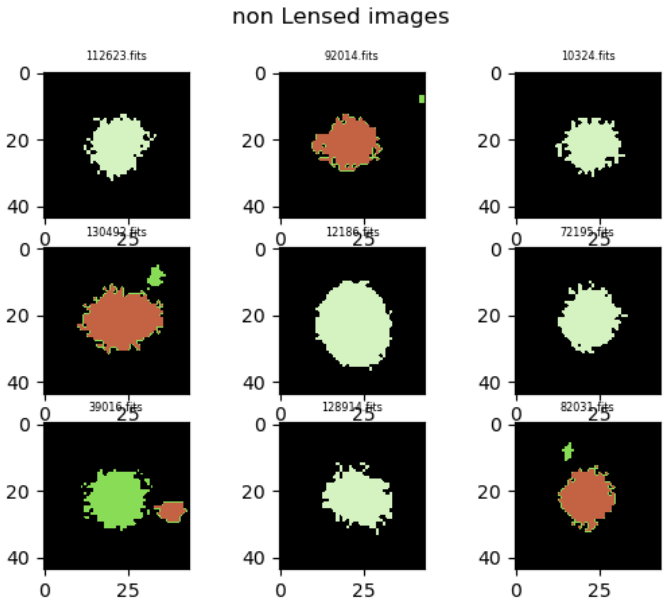

Fig. 3.1.: Images of segmentations maps for `LRG only` images of fig 2.2. A different color of segment represents a different source

function is used to dissociate 2 different sources that are sufficiently close to be detected through the same segment. This is highly interesting to report contaminating sources like a neighboring galaxy or star. However, in the perfect case of a lensed galaxy without any contaminating source, the lens would be first detected in the same segment as the deflector (as expected), but next it would be deblended from the galaxy as a different source and even sometimes the lens itself can be cut into several segments as some images from fig 3.2 illustrate it. In this case, we are facing the same problem as before, which is to identify the nature of the source, and if all the segments are part of the lens or a contaminating source. And since it's more difficult to know whether the deblended sources are lenses or contaminating sources than detecting them all together, I will omit this step and quantify the impact of close contaminating sources on the performances later. Nevertheless, a good enough segmentation should already take into account the lens images inside the segmentation map while contaminating sources that are not too close to the source of interest should be detected in another segment. Only the really close contaminating sources should be part of the main segment and interfere with the computation of our parameters. This is controlled by the detection threshold that we have to choose to minimize the effect of contaminating sources on the classifier performances. An example of a convenient situation is displayed in fig 3.3. So, we need a low enough threshold able to take into account the lens but a high enough threshold to avoid the segmentation to take into account contaminating sources.

I explored multiple methods to determine the threshold, for instance, a manual (trial and error) method, a MAD-based method, and an automatic threshold determination. The manual method is obviously not the best method but it has the advantage to give an idea of the values the threshold can take to have the expected results. The two main methods I deeply explored and tested were the automatic and the MAD-based threshold. MAD means Median Absolute Deviation and is defined as $MAD \equiv median(|X - \tilde{X}|)$ with $\tilde{X} = median(X)$ and X the dataset. The MAD will be used to compute a sharper and robust standard deviation σ by the following relation: $\sigma \approx 1.4826 \times MAD$. We expect this sharper evaluation to keep the lens while rejecting contaminating sources. The detection threshold will then be the multiplication of σ and a multiplicative factor to have a more lax or stricter condition. I used the `mad_std` function[2] from the `astropy` [39] python package which is a common package used in astrophysics. The second method I use is the automatic threshold which is a `photutils` built-in function: `detect_threshold`[3]. This function takes as input the data, and the number of standard deviation above the background flux that can be considered as a pixel possibly part of the source. In addition, it is possible to provide the value of the background and also the value of the 1-sigma standard deviation of the background or simply the RMS^2 array which is called here the error. If the background and the error are not provided, they are computed using a sigma clip. The value returned by `detect_threshold` is $\text{threshold} = \text{background} + \text{n\_sigma} \times \text{error}$ Again I will test those methods and different combinations of input parameters and address the results in the next chapter.

A last thing to build and that we will need in the next step is a mask that will identify the position of contaminating sources which are not already included in the main segmentation map if it is the case. At this step, we need a segmentation map that only contains the main source segment and a mask that contains all the other sources segments that we want to avoid during our background calculations for instance. This mask is just a boolean array: `True` for the pixels we want to eliminate and `False` for the rest. To take as an example the figure 3.3, the mask would only contain the red segment which is labeled with the `True` statement and `False` for the rest of the array while the segmentation map will in the following only contains the main green segment.

[2] `https://docs.astropy.org/en/stable/api/astropy.stats.mad_std.html`
[3] `https://photutils.readthedocs.io/en/latest/api/photutils.segmentation.detect_threshold.html`

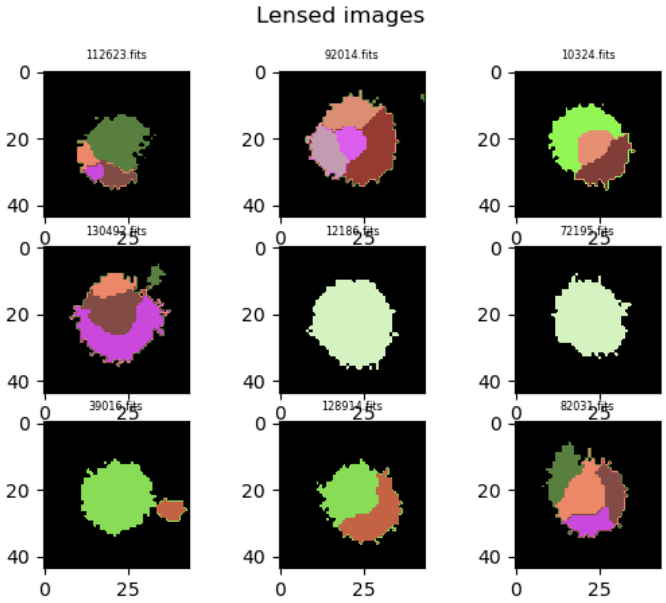

Fig. 3.2.: Sample of deblended lensed images. The deblend of the image detects the different sources that are mixed within a segmentation map. Here you can see additional segments that were found by the deblending function and that were originally included in the main segment in fig 3.1. We note that 3 lensed sources out of 9 are not deblended from the foreground source while the majority seem to be.

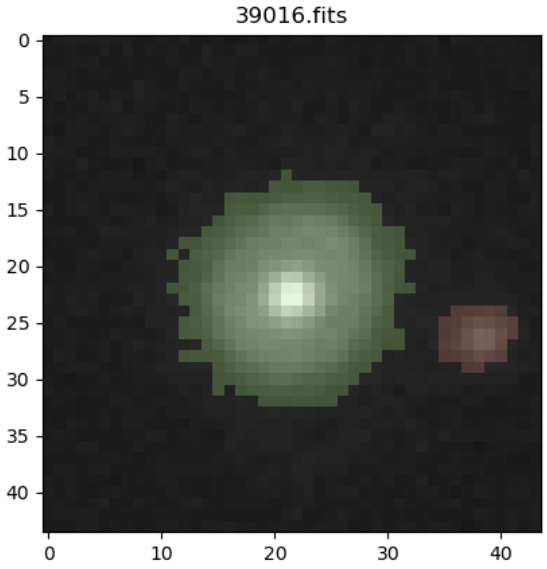

Fig. 3.3.: An example of ideal segmentation in the presence of a contaminating source. We can see both segmentation and the image in transparency. The lens ring and the deflector galaxy are well detected into a single segment, while the contaminating source is detected as another segment. The green segment represent the segmentation map while the red segment represents the mask. This example was built for illustration purposes.

3.2 Parametric and non-parametric indexes

Now that we have our segmentation map that identifies the position of the source, and a mask that identifies contaminating sources we can compute a variety of parameters and indexes. This is performed with `statmorph`[4] [40], a Python package for calculating non-parametric morphological diagnostics of galaxy images, as well as fitting 2D Sersic profiles. `statmorph` takes as input an image, the segmentation map associated, a weight map or a gain value, a mask, and the psf file. We could use a gain value but a weight map should increase performance as it associates an appropriate weight to each pixel of the image in the calculation. An easy way to build it is to take the inverse of the RMS file.

`Statmorph` provides parameters from the 2D-Sersic profile as well as morphological indexes. I choose 8 different techniques to characterize the light distribution. There is a total of 10 different parameters that will be explained in the coming subsection. This parameters are :

- Sersic parameters: amplitude, effective radius, Sersic index, ellipticity
- Asymmetry
- Concentration
- Deviation
- Gini
- Intensity
- M_{20}
- Smoothness

3.2.1 Sersic parameters

The Sersic parameters are amplitude I_e, effective radius R_{eff}, Sersic index n, center coordinates x_c and y_c, ellipticity e, and rotation angle θ. As a reminder, the Sersic profile follows this 1D-formula that can be generalized in 2D: $I(R) = I_e exp\left(-b_n\left[\left(\frac{R}{R_{eff}}\right)^{\frac{1}{n}} - 1\right]\right)$. The amplitude is the surface brightness at the effective radius. The effective radius - also known as the half-light radius - is the radius containing half the luminosity of the galaxy. The Sersic index controls the steepness

[4] `https://statmorph.readthedocs.io/en/latest/`

of the luminosity profile. The center coordinates identify the position of the center of the galaxy in the image. The ellipticity is defined as $\sqrt{1-\frac{b^2}{a^2}}$ with a the semi-major axis and b the semi-minor axis of the ellipse ($e = 0$ is a circle). And the rotation angle θ that is positive from the positive x-axis in trigonometric direction, defines the rotation of the ellipse on the image.

As discussed earlier, we mainly expect amplitude, effective radius, Sersic index, and ellipticity to be discriminant parameters. Indeed, the presence of a lens image should increase the effective radius, but decrease the surface brightness at the effective radius (amplitude). The lens should also decrease the steepness of the profile by adding light at the periphery of the galaxy and thus reduce the value of the Sersic index. Ellipticity should also be affected with higher ellipticities for non-symmetric lenses and lower ellipticities for ring lenses that should circularize the luminosity profile of the galaxy. The center coordinates and rotation angle are not expected to have any impact since there are randomly distributed variables. In addition, our sources are all centered on images.

However, parameters do not always follow the expected trend. This is the case for the 2 last examples in fig 1.5. In fact, other factors are at stake, such as the difference in the area of the segmentation when the lens is added, the magnification and thus the relative luminosity between the deflector and the lens, or also the gap between the deflector and the lens.

3.2.2 Asymmetry

Apart from Sersic parameters, `statmorph` computes a set of morphological indexes that are commonly used to describe galaxies. I made a selection of 7 different indexes that could be impacted by the presence of the lens. Let's first introduce the asymmetry index. Asymmetry [31] quantifies how rotationally symmetric the luminosity profile of the galaxy is. It is basically the subtraction between the image and the 180-degree rotated image around the galaxy's central pixel. With $I(i,j)$ as the value of the pixel in position (i,j), $I_{180}(i,j)$ the value of the pixel in position (i,j) rotated of 180 degrees about the central pixel of the galaxy, and B_{180} the average asymmetry of the background; the complete formula is :

$$A = \frac{\sum_{i,j} |I(i,j) - I_{180}(i,j)|}{\sum_{i,j} |I(i,j)|} - B_{180} \tag{3.1}$$

B_{180} is computed thanks to the average value of the absolute difference between background pixels and rotated background pixels. The central pixel is determined by minimizing asymmetry. As notified in [31], very smooth elliptical galaxies get low asymmetry while spiral, irregular, or even merging galaxies will get high asymmetry. The presence of a lens will act like an irregular galaxy and we expect a higher Asymmetry value while non-lensed galaxies will stay in lower values. In the case of a full-ring lensed galaxy, the Asymmetry should settle in the lowest values and we will maybe miss those cases. However, the combination with a large panel of parameters should compensate for this effect.

3.2.3 Concentration

Concentration [31], is the logarithmic ratio between the 80% total flux radius and the 20% total flux radius. With r_{80} the circular or elliptical 80% total flux radius and r_{20} the 20% total flux radius, Concentration is expressed as:

$$C = 5 \log \left(\frac{r_{80}}{r_{20}} \right) \tag{3.2}$$

The central pixel is also determined by minimizing Asymmetry. As the presence of a lens should decrease the effective radius, r_{80} and r_{20} should also be affected and we thus expect the concentration to be lower.

3.2.4 Deviation

Deviation [32] is a measure of distance between the intensity centroid and the center of the brightest region. The deviation is a simple Euclidean distance between the centroid coordinates x_{cen} and y_{cen} and the center of the brightest regions $x_{l(1)}$ and $y_{l(1)}$. With n_{seg} the number of pixels within the segmentation map, the deviation is expressed as:

$$D = \sqrt{\frac{\pi}{n_{seg}}} \sqrt{(x_{cen} - x_{l(1)})^2 + (y_{cen} - y_{l(1)})^2} \tag{3.3}$$

And coordinates of the centroid are evaluated with:

$$(x_{cen}, y_{cen}) = \left(\frac{1}{n_{seg}} \sum_i \sum_j i f_{i,j}, \frac{1}{n_{seg}} \sum_i \sum_j j f_{i,j} \right) \tag{3.4}$$

We expect the deviation to have greater values in the case of a highly magnified lens but unfortunately for a low magnification, the deviation will stay near 0. Galaxies with active star formation or 2 pseudo bulges can mix with the high-magnified lensed population. Even if I have low expectations with this index, I keep it in case it could be disentangled thanks to another parameter.

3.2.5 Gini

The Gini index [31][32] was first designed for economics but it can easily be used in our field of research. The Gini coefficient measures the inequality of light in the light distribution within the galaxy. This index is based on the Lorentz curve of the galaxy's light distribution and does not depend on the spatial position. With n the number of pixels involved, $\bar{X}$ the mean value of the pixels, and X_i the ith pixel value, the classical definition of the Gini coefficient is given by:

$$G = \frac{1}{2\bar{X}n(n-1)} \sum_{i=1}^{n} \sum_{j=1}^{n} |X_i - X_j| \tag{3.5}$$

Here the Gini index is computed thanks to the following equation:

$$G = \frac{1}{\bar{X}n(n-1)} \sum_{i}^{n} (2i - n - 1) X_i \tag{3.6}$$

$G = 0$ is a completely equalitarian population that corresponds to a constant luminosity profile in our galaxy frame of work. In case a pixel contains the total flux of the galaxy, Gini would be equal to 1. A lower value of the Gini index is expected with the presence of a lens. Indeed, the lens should input additional light in the peripherical regions that are in general the faintest regions. This has the effect of reducing inequalities in the luminosity profile.

3.2.6 Intensity

The Intensity index [32], is the ratio in flux between the two brightest regions of the galaxy $I = \frac{I_{(2)}}{I_{(1)}}$ with $I_{(2)}$ and $I_{(1)}$ the second and first brightest regions fluxes. The evaluation of these two fluxes is performed by smoothing the luminosity profile with a Gaussian blur. The region is then defined by following the maximum gradient path with the calculation of 8 neighbor pixels. The process stops at a local gradient maximum and the region is made of pixels linked to the maximum. Galaxies with a single bright bulge approach 0 while lensed galaxies can host a secondary bright region and approach values close to 1. However, the same remarks as the Deviation index can be made but I keep it in case a disentanglement can be made thanks to another index.

3.2.7 M_{20}

The second-order moment is the flux of each pixel multiplied by the squared distance to the center of the galaxy. We use the M_{20} index [31][32] which is defined as the normalized second-order moment of the 20% brightest pixels. By ranking pixels in order of decreasing flux, the pixels moment is summed while the total of the summed brightest pixels equals 20% of the total flux:

$$M_{20} = log\left(\frac{\sum_i M_i}{M_{tot}}\right), \quad \text{while} \quad \sum_i f_i < 0.2 f_{tot} \tag{3.7}$$

With M_i the second order moment of the ith pixel, f_i the flux of the ith pixel, f_{tot} the total flux, and M_{tot} the total second order moment :

$$M_{tot} = \sum_i^n f_i \left[(x_i - x_c)^2 + (y_i - y_c)^2\right] \tag{3.8}$$

The central coordinate x_c and y_c are determined such that M_{tot} is minimal. The M20 index is more sensitive to mergers and also active star-forming regions in the galaxies. This index is more sensitive than the Concentration index for these two types of galaxies because it does not take a circular or elliptical aperture and the galaxy center is a free parameter. It is expected to have higher M_{20} values when a lens is present because the second-order moment of a lensed pixel is larger than a pixel with the same flux in the bulge because of the distance to the center. Again, low-magnified lenses can be misclassified with non-lensed galaxies, because the flux

of the lens is not high enough to be in the 20% brightest pixels. Ring galaxies can also settle in high M_{20} values.

3.2.8 Smoothness

Finally the smoothness [31] index quantifies the number of small structures. It is the substraction of the smoothed image to the original one. The definition equation is given by:

$$S = \frac{\sum_{i,j} |I(i,j) - I_S(i,j)|}{\sum_{i,j} |I(i,j)|} - B_S \tag{3.9}$$

With $I(i,j)$ the pixel intensity at position (i,j), $I_S(i,j)$ the intensity of the pixel at position (i,j) of the smoothed image, and B_S the average smoothness of the background. This index seems to work for high spatial-resolution images. We know it works for well-resolved images, and the smoothness should increase in the presence of a lens. Indeed, lenses add small structures at the periphery of the galaxy that can be found in higher values of smoothness.

3.3 Classifiers

After dimensionality reduction, the next step is the classification. Although, We substantially reduced the dimensions of the dataset but we still have a multidimensional parameter space. In the parameter space, the distribution of these parameters piles up and spread on different regions depending on the class of the object (if lens or not). Sometimes both distributions can overlap in some dimensions while they are detached from each other in other dimensions. This segregation between clusters of items is a chance for us to differentiate lensed cases from non-lensed galaxies. This is where classifiers play a major role, they automate the computation of a multidimensional border that allows us to identify the category of the galaxy. There are many classifiers algorithms that we tested but I will mainly focus on 3 main algorithms: Support Vector Machine (SVM), Random Forest, and Multi-Layer Perceptron (MLP). Before introducing the algorithms, it is important to mention that I use a widely used Python package: `scikit-learn`[41]. It is a massive library of machine-learning tools and where all the algorithms I used were implemented.

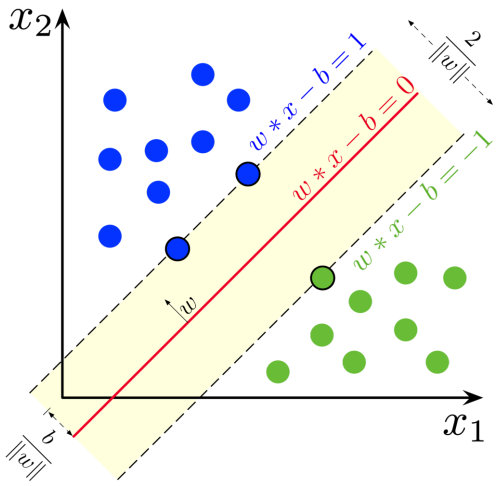

Fig. 3.4.: Representation of the support vector machine algorithm in a 2D parameter space. Class blue and class green are separated by the red hyperplane. By Larhmam[5]

3.3.1 SVM

First, let's explain the Support Vector Machine algorithm. In order to have a better comprehension of this classifier, a simple classification in a two-dimensional space is displayed in fig 3.4. This algorithm [42][43] is built with the simple idea of separating two different classes of parameters by a hyperplane (a line in 2D, a plane in 3D) such that the distance between the closest points and the hyperplane is the highest. This hyperplane is represented by the red line in fig 3.4 and the yellow zone is called the margin. The so-called "support vector" is the closest data from the hyperplane. In case the limit is not a simple hyperplane, we use the kernel trick. This trick allows the use of a linear classifier in non-linear situations. The parameter space is transformed into a higher dimension parameter space that allows a better separation by a hyperplane. Figure 3.5 is an illustration of the kernel trick.

3.3.2 Random forest

The second algorithm is the random forest. This algorithm [43][44] is in fact a combination and an optimization of a simpler algorithm which is the decision tree so let's explain the decision tree algorithm. The decision tree classifier is able to make a classification from a succession of decisions that can be represented by a tree as it is illustrated in fig 3.6. The random forest is just an ensemble learning method combined with Bootstrap. The ensemble method used in the random forest algorithm is the bagging that consists in combining multiple Decision trees like a committee of decisions. During the process of tree building, only k attributes are

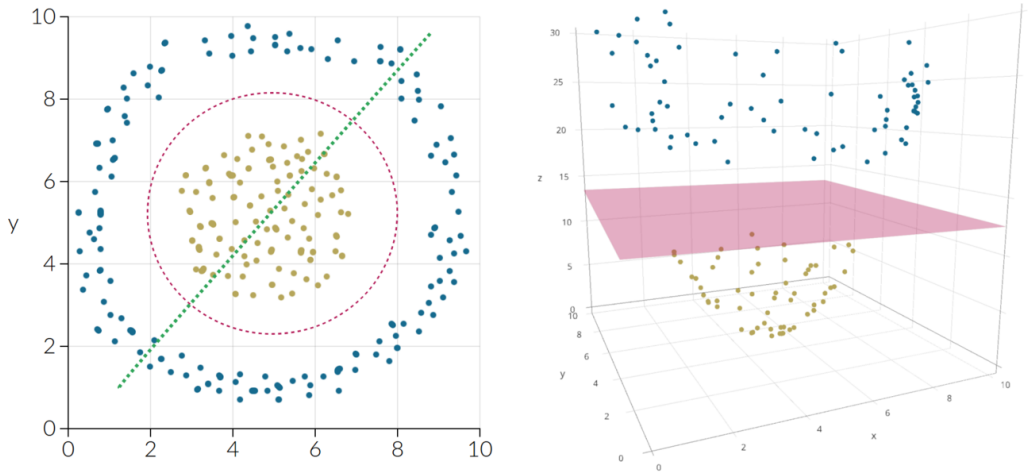

Fig. 3.5.: Representation of a 2D distribution (left) and the reparametrization of this distribution into a 3D space (right). The green dotted line represents the linear SVM while the red dotted line is associated with the red plane and represents the SVM using the kernel trick. The green line does not succeed to separate the two distributions in a convenient way, while the red plane (and thus the red circle) perfectly classify the distribution. By Wikimedia[6]

selected out of n attributes in the random forest algorithm. Each decision tree gets a random subsample of the dataset (with replacement) as a learning dataset, this is the bootstrap method. Each classification made by the committee of decision trees is stored and the final class is attributed with a majority vote. An illustration of the random forest is available in fig 3.7.

3.3.3 Multi-Layer Perceptron

Finally, the third classifier: the Multi-Layer Perceptron. An MLP [43][45] is a fully connected network of artificial neurons, called perceptrons. An artificial neuron is an elementary unit designed to receive one or more inputs that will be linearly combined with appropriate weights to return an output. Weights are learned by the algorithm to fit the wanted result. The output of the neuron is usually transmitted to an activation function (sigmoid, tanh, ReLU...) that will ensure a normalized output. A Multi-Layer Perceptron is a network of these elementary units arranged in layers of a few neurons. The architecture of this network is displayed in figure 3.8, the first layer is the input layer that receives the data, next several fully connected layers can follow along and are called hidden layers, and finally, we have the output layer. While a single neuron can handle a simple function, an MLP can deal with highly non-linear functions with a combination of multiple artificial neurons. In our case, I expect the MLP algorithm to return the best results, because such a method allows

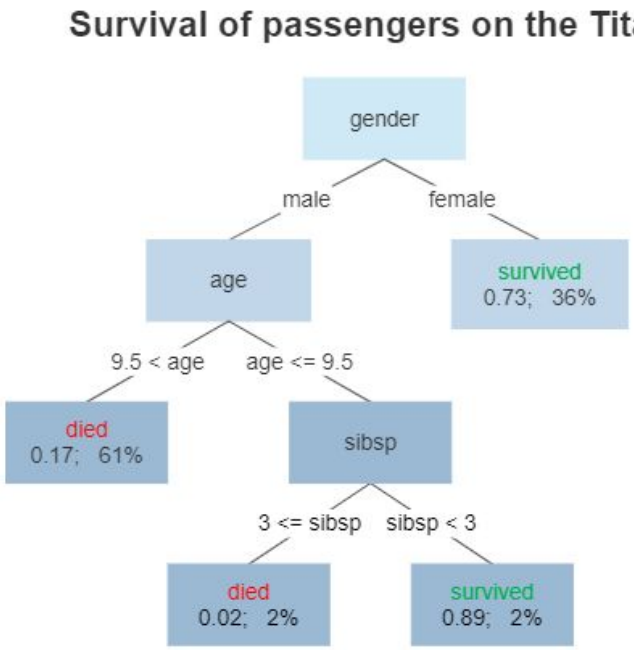

Fig. 3.6.: Decision tree that classifies survivors of the Titanic wrecking. Here we clearly see that survivors were ladies and children less than 9.5 years old with less than 3 siblings. By Wikimedia[7]

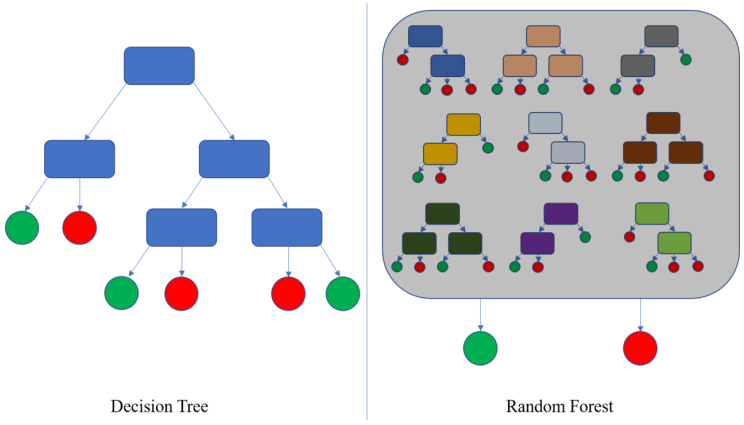

Fig. 3.7.: Example of a decision tree architecture (left), and architecture of a random forest algorithm (right). The gray zone represents the committee of decision trees trained on Bootstrapped datasets. The final class is attributed to the majority class attributed within the committee. By Wikimedia[8]

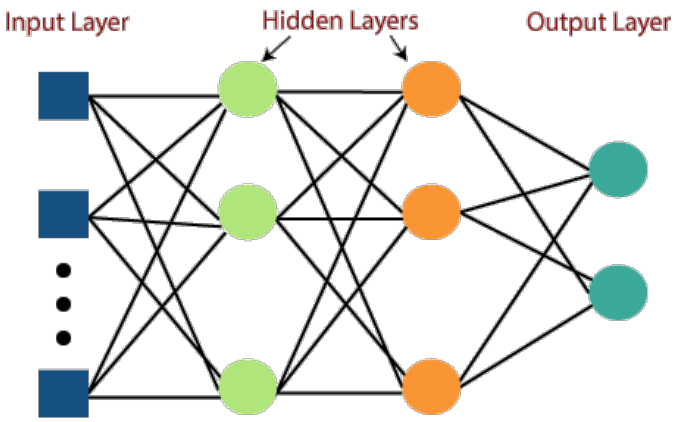

Fig. 3.8.: Architecture of a Multi-Layer Perceptron (MLP)

us to have more complex functions to separate the 2 distributions. Indeed, artificial neural networks are considered Universal approximators, which means that they can be viewed as a more sophisticated version of Taylor's series expansion.

3.3.4 Performance scores

Performance of these classifiers will be presented in the chapter 4 but we first have to enumerate our requirements to identify what is a nice result. Obviously, a perfect classifier that returns all the lenses present in the data without any misclassification would be the best result. In reality, we will not have a perfect classifier so we want it to return a maximum of lenses with a low rate of false positive cases. To evaluate the performances of our classifiers we will divide our dataset into a training set and a test set. The training set represents 60% of the total size of the dataset and the test set is 40% of the total size. The training set is obviously reserved to train the model while the test set is reserved to make an unbiased evaluation of the model. Since positive cases are lensed galaxies (P) and negative cases are non-lensed galaxies (N), the True Positive instance is thus the number of lensed galaxies that are recovered by the classifier as lensed galaxies. True Negative are non-lensed galaxies recovered as non-lensed galaxies, False Positives are galaxies that are recovered as lensed galaxies but are not lensed in reality. Finaly, False Negative are lensed galaxies that are recovered as non-lensed ones. When the evaluation is performed we get the confusion matrix which summarizes the number of True Negative (TN), False Positive (FP), False Negative (FN), and True Positive (TP) predictions:

$$\begin{pmatrix} \text{TN} & \text{FP} \\ \text{FN} & \text{TP} \end{pmatrix}$$

Performances can be measured with some commonly used quantities: accuracy, precision, and recall. The accuracy is the fraction of correct predictions. The accuracy is a quantity that reflects how good the predictions of the classifier are. The precision returns the fraction of correct positive predictions out of the total positive predictions. A low precision reflects a high amount of false positive predictions. The recall returns the fraction of correct positive predictions out of truly positive ones. It reflects the ability of the classifier to find all the positive cases. Precision and Recall are sometimes called purity and completeness for these reasons. All these numbers are computed thanks to the confusion matrix that returns the expression of accuracy, precision, and recall by :

$$Accuracy = \frac{TP + TN}{P + N},\ Precision = \frac{TP}{TP + FP},\ Recall = \frac{TP}{TP + FN} \tag{3.10}$$

With $P = FN + TP$ and $N = FP + TN$. The two main quantities we want to maximize are precision and recall. High precision is required in order to have the fewest false positive case and then avoid waste of observation time on following-up telescopes. And then a high recall is interesting because we want to identify the maximum of lenses present in the dataset. To synthesize these performances in one single index, it is interesting to consider the F-score which is the harmonic mean between precision and recall.

$$Fscore = 2\frac{Precision \times Recall}{Precision + Recall} \tag{3.11}$$

It is also important to emphasize that accuracy and precision are highly sensitive to prior. For example, a completely unbalanced dataset will massively impact accuracy and precision while recall is not affected. In our case, we have a 50/50 dataset.

4 Results

This chapter is exclusively dedicated to the results of my work and what could be understood from them. As exposed in the previous part in section 3.2, we want to maximize the precision, the recall, or just the f-score. In order to get the best results, we will try to find the right input parameters and thus have the highest scores. So we will choose our input parameters such that these quantities reach their maximum values. However, failing cases can happen, for whatever reasons that I still not understand, `statmorph` is not always able to process images. This means that no data is retrieved from images. Hopefully, this represents around 10% of the dataset and the vast majority is processed. 10% of missing data is anyway a significant part and we will discuss later solutions that can be implemented to handle this problem.

In addition to that, the Sersic model fit is not always able to converge or the computation of morphological indexes can be considered as non-accurate enough. In that case, a value is returned anyway but a flag is associate to it. There is two flag parameter, one for the quality of the Sersic fit, and another one for the quality of the morphological indexes. In the case of the Sersic model, it is a binary flag (0: no problem, 1: convergence problem). For morphological indexes, the flag can take up to 4 values according to the severity of the computation (0: good, 1: suspect, 2: bad, 4: catastrophic). In order to get better results I will introduce those flag values in the parameter matrix. I will nevertheless propose solutions to reduce flagged cases and increase classifiers scores.

4.1 Parameter space

In this section, the input parameters will be fixed but their influence on the results will be discussed later in section 4.4. Values of our input parameters are fixed to values that maximize the F-score. Now that our segmentation parameters are fixed, `statmorph` provides us with a parameter space that ensures good performance. The parameters are described in section 3.2. Again the parameter space is a high-dimensional space that can't be seen as a whole within an image. To have a better

idea of the behavior of indexes in the presence of a lens, we can plot them by pair of parameters. The distributions are scattered in fig 4.1. Before discussing further the results of these plots, it is important to know that these figures only represent perfectly working cases, flagged points follow the same behavior and are established in the same regions of the parameter space. To not overload the graphic, we only consider perfectly working cases. As revealed by the scatter plots of amplitude vs effective radius and n vs ellipticity, Sersic parameters are not the best at dissociating lensed from non-lensed distributions. However Morphological indexes seem to well differentiate the two classes. The clearer separation between non-lensed and lensed distributions is the asymmetry and the concentration indexes as can be observed in fig 4.1. Other striking plots can be found in the appendix A of this work.

As already discussed the expected trend on Sersic indexes is not always respected due to external reasons. It can be well observed since lensed and non-lensed point distributions are massively overlapping each other. As fig 4.1 suggests, the combination of the Sersic index with the ellipticity is a better-discriminating combo. Nonetheless, the separation is not that obvious and can still produce misclassification. On the other hand, asymmetry, concentration, Gini, and M_{20} indexes seem to be the most powerful combination of indexes.

In addition to these parameters, I will use a quantity that is computed by `statmorph` and that can help the classification. This quantity is the Signal-Noise Ratio (SNR). This quantity, as its name suggests is the ratio between the signal (source pixels) and the noise (RMS of the same pixels). Here `statmorph` returns the mean value of this ratio. Regarding the figure 4.2, the SNR is higher on average for lensed images which is coherent with the design of the lensed images. Indeed, lensed images have an extra signal from the lens image that we added to the foreground galaxy.

The last parameters that will join the parameter matrix are the flag values that can play a role in the classification. Indeed, we have slightly more flags for non-lensed galaxies (53% of the non-lensed galaxies) while 49% of the lensed galaxies are flagged. This can be a nice indicator for some difficult cases. Let's see next how the classifier scores are affected by the presence or not of some parameters. Referring to the table 4.1, we clearly see the importance of the morphological parameters in the classification process. We can also notice better performance using the SNR. The fact that the SVM is better without the Sersic parameters is understandable since those parameters are largely intricated. And finally, we observe that without flags parameters, the SVM is able to get the highest performances. Including the flags, maybe is too sophisticated for a SVM. Maybe these parameters will be better accepted in the random forest classifier since it is a model following sequential

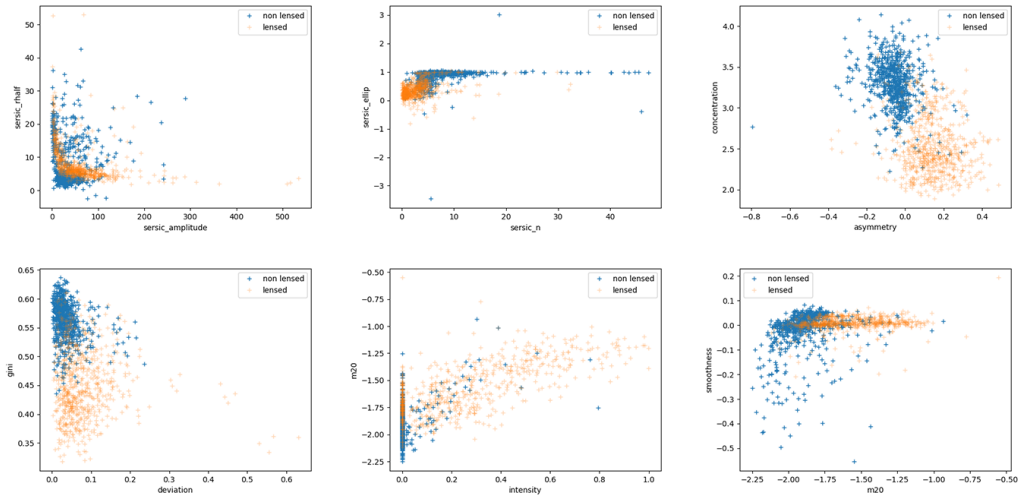

Fig. 4.1.: Representation of the 11 parameters in 6 different scatter plots.

	Accuracy	Precision	Recall	F-score
all	92.00	93.15	92.07	92.61
w/o flags	93.36	93.54	92.46	93.00
w/o SNR	92.16	91.54	92.07	91.80
w/o Sersic	93.27	93.36	92.46	92.90
w/o morph.	86.54	86.53	84.53	85.69

Tab. 4.1.: Scores of an SVM classifier depending on the parameter matrix

decisions. SVM is a classifier attempting to separate classes by hyperplanes that represent a smooth and continuous border. This implies approximations in the border that can require a non-continuous border. Decision trees can easily create such discontinuities. MLP classifiers can also model more complex borders so flags should be more interesting in the case of both random forest and MLP.

4.2 Classifiers

In this section, we will discuss the performances of all 3 classifiers with their variations. For instance, SVM can handle multiple kernels so we will test the different performances. We will then compare the best SVM to the random forest and MLP. We will also compare the performances without some types of data like in the previous section. The results can be found in table 4.2.

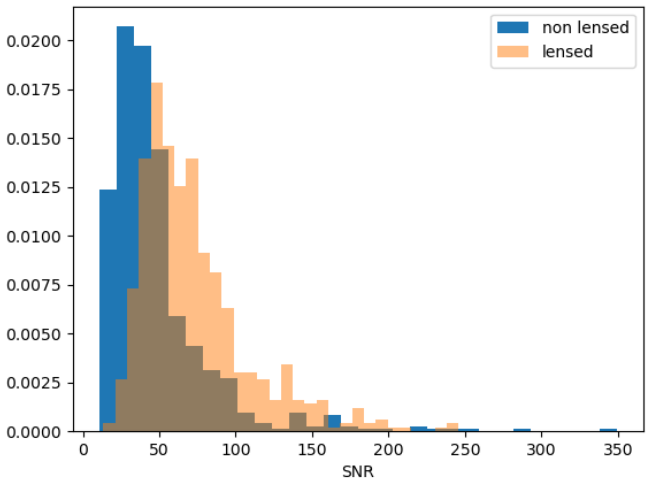

Fig. 4.2.: Distribution of the Signal-Noise Ratio (SNR) depending on the class of the image.

Let's begin with the SVM. We already saw that the best result is found without flags included in the data. so I kept this dataset and tested SVM with 4 different kernels. The worst result is given with the sigmoid kernel. With the polynomial kernel the recall, and the f-score fall below 90% even if the precision reach 95%. Even if the precision is high the recall drop at 85% and the RBF kernel ensures better global scores (F-score). It is important to note that this polynomial is the best polynomial which is a third degree while lower and higher polynomial degrees make the scores drop even lower. Because the RBF kernel is the default one we get back the same result as in the previous section which is the best result found with the SVM classifier.

Next, the random forest outperformed the SVM performances as the table 4.2 report it. The improvement is only a few tenths of a percent but still an improvement. As predicted before, the flags indexes have a positive effect when taken into account. Still, this effect has a limited impact. We again verify the importance of the SNR and the morphological parameters on the results. To summarize, the random forest gets its highest scores with all the set of parameters as we predicted before.

Finally, the MLP is the method that reports the best scores (see table 4.2). It still improves by a few tenths of percent and even a percent in the case of precision. It uses the ReLU activation function. We can easily make the same remarks as for the random forest classifier since the best result is given with all the set of parameters. We can just notice that flags have an even lower impact on the performances than in the random forest case. Even if the differences are not significant between "all" and "without flags", it is preferable to have higher precision over the recall so we will

consider the MLP with all the set of data, the best classifier of this study. To have faster results, I used for each method a subset of the total data which represents a sample size of 3000. This model gave the following confusion matrix :

$$\begin{pmatrix} 546 & 22 \\ 36 & 481 \end{pmatrix}$$

4.3 Study of the false positives and false negatives populations

To have a better view of the difficulties encountered by the MLP to classify some cases, we will scrutinize the false positives and false negatives classified images. In figure 4.3 we can identify 3 main causes of the misclassification as lensed galaxies. The first one is the presence of one or several close contaminating sources. It can be a merger. The second one is the presence of a high noise in the image and the outer regions of the galaxy are hidden in the noise, and the noise is then considered as the lens component. This results in a large bulge that can be classified as a lens due to an increase in the effective radius for instance. In general, the Sersic parameters are modified but seem to be close to a blurry classification border. The third reason is that the galaxy is highly elliptical and has a large and luminous disk.

In false negative cases in fig 4.4, the main reason is the presence of a large gap between the deflector and the lens image which is not detected inside the main segment. The second reason is that the flux of the lens images is too low and is mixed with the background.

4.4 Impact of input parameters

In this part, I will explore all the results associated with the segmentation map and the impact it has on results. We will discover the sensitivity of the performances to the input parameters of the detection functions.

	Accuracy	Precision	Recall	F-score
SVM				
linear	92.07	92.84	90.33	91.57
polynomial	90.88	95.63	84.72	89.85
sigmoid	85.71	86.79	82.59	84.64
rbf	93.36	93.54	92.46	93.00
random forest				
all	94.29	94.87	93.04	93.94
w/o flags	94.01	93.46	94.00	93.73
w/o SNR	93.18	93.01	92.65	92.83
w/o Sersic	93.46	93.55	92.65	93.10
w/o morph.	89.03	89.48	87.23	88.34
MLP				
all	94.65	95.63	93.04	94.32
w/o flags	94.38	95.78	92.26	93.99
w/o SNR	93.82	95.36	91.49	93.39
w/o Sersic	93.73	94.81	91.88	93.32
w/o morph.	88.02	88.92	85.49	87.18

Tab. 4.2.: Scores of multiple classifiers with variations on the dataset

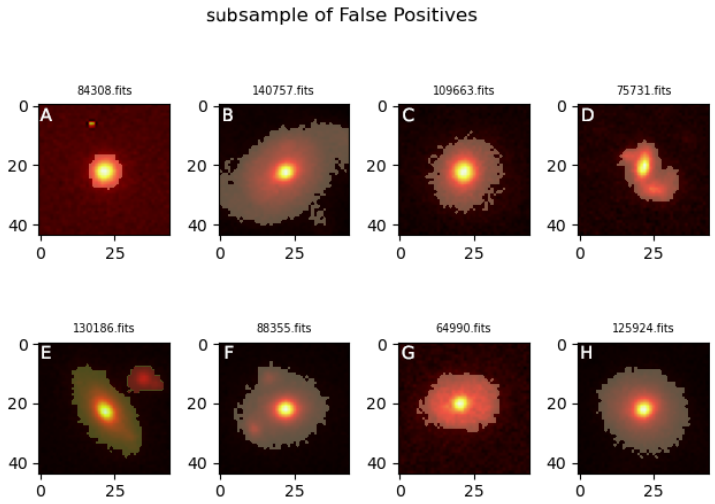

Fig. 4.3.: A sub-sample of 8 false positive images. There are mainly 3 scenario: (1) it is a merger galaxy or there are one or many close contaminating sources (panels D, F), (2) there is a large bulge or the outer part of the galaxy is mixed in the background noise (panels A, C, G, H), (3) the galaxy is highly elliptical and/or have a luminous disk (panels B, E). Segments are displayed on top to identify the detection.

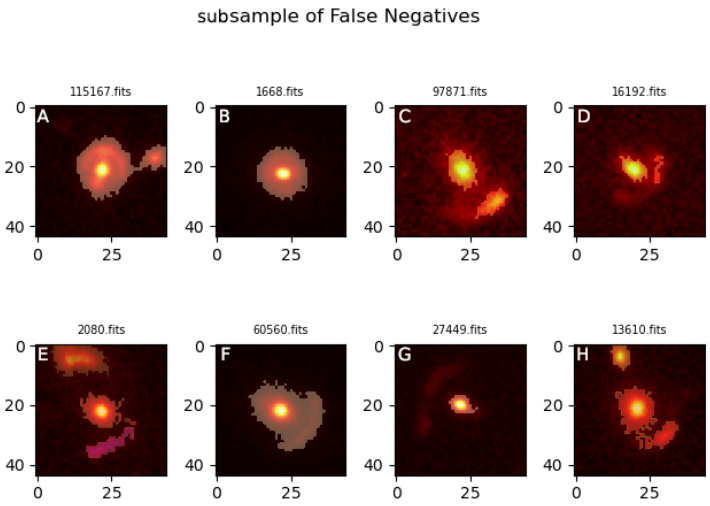

Fig. 4.4.: A sample of 8 false negative images. There are 2 main scenarios: (1) the gap between the deflector and the lens is too large to be detected in one single segmentation map, (2) the flux of the lens is too faint and is mixed in the background noise. Segments are displayed on top to identify the detection.

4.4.1 `Detect threshold`

From the image and the number of sigmas `n_sigma` above which the function `detect_threshold` performs a source detection, this function is able to determine a convenient threshold value adapted to each image. So, it is important to quantify the impact of the different input parameters on the performances. In all our tests the background mean value is evaluated with the corner method, which was already discussed in part 2.3. For the error parameter, I did 2 tests with two different methods: (1) the evaluation of the error is computed by the standard deviation of the background using the corner method and (2) the RMS^2 file is given as a 2D array of errors for each pixel. It is also important to know that our results are performed with the default SVM classifier (with RBF kernel) which we will consider as our fiducial classifier. Different tests on the classifiers will be performed later.

We have three main cases of data, the first one is the data without any problem, where the Sersic fit as well as the morphological indexes can be trusted. The second case is the case where the Sersic fit is not converging and has to be considered with precautions. This case is reported by a Sersic flag equal to 1. The third class is the case where the morphological computations are flagged (ie: 0,1,2, or 4). Note that both Sersic flags and morphological flags at the same time is a possibility. The last case is the case where no data is provided by `statmorph`, we call them failing or defaulting cases. By looking at figures 4.5,4.7, and 4.6, we clearly notice that failing

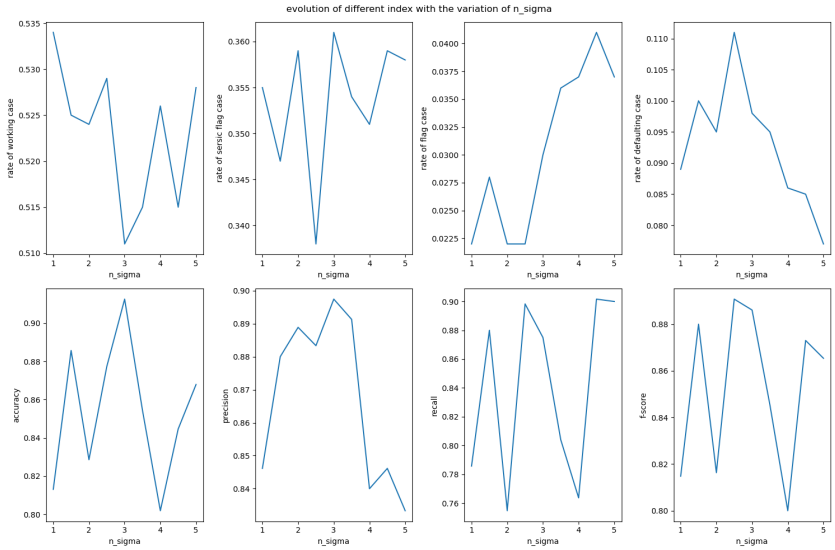

Fig. 4.5.: Performance of the fiducial SVM classifier with the `detect_threshold` method, mean and standard deviation of the background using the corner method. In the top panel, we can see the evolution of `working cases`, `Sersic flag cases`, morphological `flag cases`, and `defaulting cases` depending on the value of `n_sigma`. The bottom panel represents the evolution of the different performance scores: accuracy, precision, recall, and the F-score. To maximize the score values and in particular the precision score, and minimize the number of flags and failing cases, a good value of `n_sigma` would be 1.5. This value is a good tradeoff between good precision and a low number of failing and flagged cases.

cases settle around 10%. while the number of morphological flags stays below 5%. The working cases are slightly above 50% and the Sersic flagged cases are rarely above 38%.

In figure 4.7, we see the test using the MAD to determine the detecting threshold. This method has the advantage of only using the data, no mean background and error need to be given. We clearly see that the scores are about the same order of magnitude, however, the results are higher in fig 4.6 so the `detect_threshold` method with RMS^2 files. The method using the corner method for both the background value and the error (fig 4.5) gets sensible same results as in fig 4.6 but slightly lower. No matter the method which is used there is a clear necessity of tradeoff to have good values in working rate and good F-score (see section 3.3.4 for the definition of the accuracy, the precision, the recall, and the f-score). In fig 4.6, with $\mathrm{n\_sigma} = 3.5$, the F-score, the accuracy, and the precision are the highest. But at the same time, the number of working cases are correct, and the number of morphological flagged cases is still low whether it reached one of its highest value. The number of failing cases is still high but it is a typical value compared with other methods. Finally, the number of Sersic flags is reaching a minimum. The recall is not

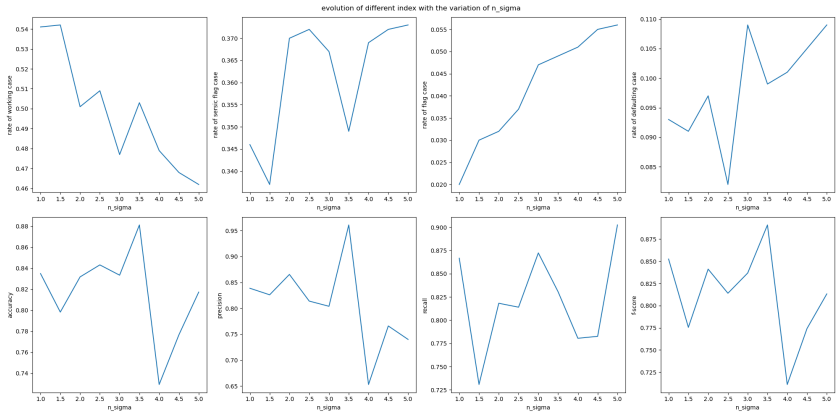

Fig. 4.6.: Performance of the fiducial SVM classifier with the `detect_threshold` method, the mean background is computed using the corner method and the error is RMS^2 file. In the top panel, we can see the evolution of `working cases`, `Sersic flag cases`, morphological `flag cases`, and `defaulting cases` depending on the value of `n_sigma`. The bottom panel represents the evolution of the different performance scores: accuracy, precision, recall, and the F-score. To maximize the score values and in particular the precision score, and minimize the number of flags and failing cases, a good value of `n_sigma` would be 3.5. This value is a good tradeoff between good precision and a low number of failing and flagged cases.

at its highest value it is preferable to maximize the precision first and next the recall since we want the lowest false positive rate. It seems that the `detect_threshold` method with the RMS^2 files and $mathrmn\_sigma = 3.5$ is the best compromise so I will use these parameters in the following of this work.

4.4.2 `n_pixel`

The last parameter that can influence the segmentation map is the number of connected pixels `n_pixels` above the threshold required to be part of the source. This parameter step in the `detect_sources` function. Fig 4.8 reveals the result of the test performed with the last best parameters. It seems that the best results (ie: high accuracy, precision, recall, and F-score) are performed at `n_pixels`=4. With a value of 4, we have a rather high rate of working cases while flagged and defaulting cases stay rather low. We also have the best scores with this value. We will then use `n_pixels`=4 in the following of this work in order to maximize the quality of the parameters.

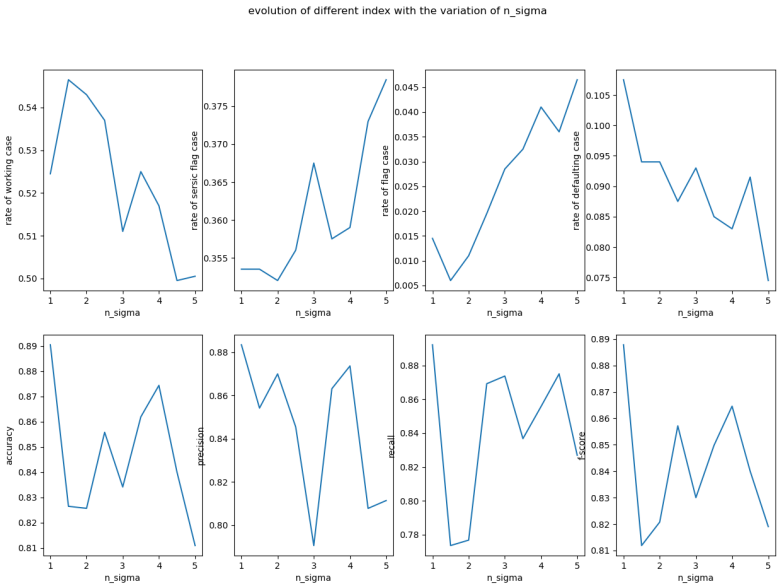

Fig. 4.7.: Performance of the fiducial SVM classifier with the MAD method. In the top panel, we can see the evolution of `working cases`, `Sersic flag cases`, morphological `flag cases`, and `defaulting cases` depending on the value of `n_sigma`. The bottom panel represents the evolution of the different performance scores: accuracy, precision, recall, and the F-score. To maximize the score values and in particular the precision score, and minimize the number of flags and failing cases, a good value of `n_sigma` would be 1. This value is a good tradeoff between good precision and a low number of failing and flagged cases.

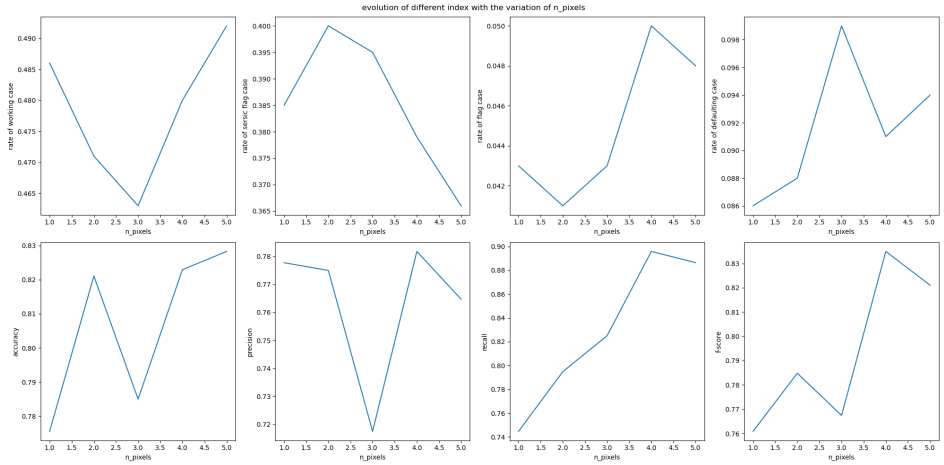

Fig. 4.8.: Influence of the `n_pixels` parameter in the `detect_sources` on the performances with the fiducial SVM classifier. `n_pixels` controls how many connected pixels have to get a higher value than the detection threshold to be considered part of a source.

Discussions & conclusion

5

The analysis in the previous chapter has enabled us to find that :

1. Thanks to an ensemble of systematic tests, we have been able to identify the best setup for optimal object detection,
2. Among multiple parameters (Sersic parameters, morphological parameters, Signal-Noise ratio, Flag values) morphological indexes are the most relevant parameters that enable a rather clear separation between the different classes, in particular asymmetry, concentration, and gini,
3. We have compared 3 different classifiers (Support Vector Machine, Random Forest, Multi-Layer Perceptron) and identified the Multi-Layer Perceptron as the best classifier mainly based on precision and recall metrics.

5.1 Discussion of the results

First of all, the results are encouraging since this work is original in its approach and the results were uncertain since no existing work using this approach has been found yet. Even if we know that contaminating galaxies within the segmentation map are more likely to arise as a false positive, we cannot quantify this phenomenon. Is it systematically classified as a false positive lens? To have a better view of this effect, it would have been nice to have the entire control on the dataset design. We could have quantified the necessary conditions to detect 2 close galaxies in a single segmentation. This can be estimated with a Monte Carlo method which chooses random coordinates, random size, and random luminosity for the contaminating source, and then the detection is performed. After that, it is easy to determine the minimum necessary conditions to have a detection of 2 close galaxies in a single segment.

In the frame of a PhD work, with longer-term deadlines, it could have been interesting to build my own dataset to get higher control over it. Having access to E.Savary's dataset was a real gain of time but also a nice data processing exercise. Again, the control on the dataset would have allowed us to get access to wider

image stamps. Indeed, too small image stamps can result in too large segmentation maps and thus too few backgrounds that impact the computation of the Gini index for example. This is an advantage of my method compared to a CNN method that accepts preformatted images in a given size. our method can easily scale to different size of images.

Earlier, I criticized the lack of interpretability of CNNs, but our best performances are based on an MLP which is also a neural network. However, we can easily draw the probability density that the MLP computed by meshing the parameter space and we can thus have a better comprehension. In fact, it is easier to understand why given variations of parameters betray the presence of a lens, while CNN returns image features that don't have a physical explanation which is one of the main goals of this work.

Now it's time to compare our results with the results from E. Savary et al. [28]. The paper reports a precision of 100% and recall of $\equiv 95\%$. This means that almost or even no false positive lens galaxy is returned by the CNN and about 95% of the lenses available in the dataset will be detected. To take the example of the Euclid mission, we expect to discover about 280,000 lenses, and the use of such an algorithm would rise approximately 266 000 quasi-certain lensed candidates. Our model is a bit less efficient than the CNN model with precision and recall both at 94.4%. The values of precision and recall are always in the same order of magnitude, while CNN allows more fine-tuning of the parameters to increase or reduce the precision or recall value. In addition, this value does not take into account the fact that we have 9% of our dataset that is discarded by `statmorph` errors, and it has to be taken into account. By considering this 9% going back in the negative cases, accuracy = 94.6%, precision = 95.6% and recall = 91.8%.

5.2 Other methods

During the process of research, many other methods came to our mind and I think they are worth exploring them.

The first method is inspired by a method used in the direct imaging exoplanetology field. Direct images of exoplanets are preprocessed such that the host star light is canceled from this image. After this preprocessing it still remains spatially and temporarily varying thermal noise that results in speckles. The exoplanet signal is thus hidden inside this thermal noise between those speckles. To reduce this thermal noise closer to the background limit, the noise can be identified by its principal

components using PCA [46] and then subtracted from the original image. PCA stands for Principal Component Analysis [47] and consists in transforming correlated variables into uncorrelated variables in order to describe the new parameter space with fewer components (the principal components). By applying PCA to the whole image, we could describe lensed and non-lensed images with 2 different sets of principal components that will allow us to categorize them. In fact, we can apply this method to the parameter space that we computed in this work. You will find in the sect A 2 graphs representing the 3 principle components of our parameter matrix. It results in still intricated distributions.

The next innovative method is to use the azimutal profile of the galaxy. The azimutal profile as I defined it, is the maximum of pixel values on a circle at a growing radius as already appeared in fig 1.5. One could have taken the average value, the sum, or the standard deviation for instance. When a lens is present, a bump appears on the azimutal profile. These different curves can maybe be recognized by one of the classifiers. Next, a PCA will give the principal components of the non-lensed profile. The transposition of the lensed profiles in the new vectorial space could result in a more distinct separation, that will be processed by one of the classifiers. One can also use the distance between the Sersic profile and the azimutal profile as an index. We can define this metric in many different ways thanks to the maximum Euclidean distance, the maximum difference along the flux axis, or the difference between the area under the curves. Again it is also possible to use all the mentioned quantities in the same time to process them in classifiers.

And finally, the last method consists in subtracting a luminosity profile from the original image and studying the residual image. In the presence of a lens, we should enhance the lens component in the residual image. This can be done by fitting a Sersic profile, and an example is present in section A. By using the morphological indexes on the residual image, the separation between lensed and non-lensed distribution in the parameter space should be enhanced and classifiers would better classify. But another technique is to train an autoencoder on images and identify lensed and non-lensed distributions in the latent space, which maybe are better segregated. An autoencoder is a neural network built from an encoder and a decoder. The role of the encoder is to represent the image in a smaller dimension space called latent space. The decoder generates the wanted image from a latent space vector. Instead of using a traditional autoencoder which encodes images into a set of low-dimension vectors that stores the most relevant data, we can use a variational autoencoder, which encodes the image into low-dimension distributions defined by only 2 parameters (μ, the mean value of the distribution and σ the standard deviation), next the discrimination between the 2 classes can be done directly in the

latent space instead of the parameter space computed in this work. Note that this method removes interpretability since the latent space is an abstract space designed with highly non-linear functions built by neural networks.

5.3 Improvements

This work is still preliminary work that can be upgraded. We can thus expect better results. We will then expose improvements that could have been done.

First, the use of the deblending source function discussed in sect 3.1 can be interesting to use but would need a deeper comprehension of the segmentation map. It also requires building a strategy to really identify separately the differences between a lens and a contaminating galaxy or merger.

The next improvement that can be done, is the use of ensemble methods and bagging methods. by combining predictions of multiple base estimators, ensemble methods improve the robustness and the generalization compared to a single estimator. The bagging improves the stability, the precision and reduces the variance of the estimator by aggregating multiple bootstraped subsamples of the original dataset. It is already used with the random forest because it is a well-known and already implemented method, but similar techniques can be built. The combination of the three different methods in an ensemble method can also improve our scores since different algorithms can make different predictions for the same image and result in completely different false positive distributions that can be crossed-checked.

Instead of improving this method and the CNN performances separately, it could be interesting to assemble these two algorithms to build a more robust ensemble method. This would be interesting to compare the misclassified populations of both techniques to compare the weaknesses of each method. The combination of both methods could counter-balance the weaknesses of the other one.

Finally, an obvious improvement is to reduce failing computations by `statmorph`. This represents 9% of the dataset which is 852 more images.

5.4 Conclusion

To conclude this work, I would say that the reached performances combined with the many improvement perspectives positions this technique as a promising one.

This technique has the advantage of making the classification less mystical than the deep learning-based methods. Indeed, the classification is based on physical and geometrical considerations that are easier to understand. The next important step of this work is to evaluate the performance of this method on real data. Indeed, those results are theoretical results that are still specific to the simulated images. It could be interesting to make transfer learning to real data and fine-tune our model. All the deep learning methods are working well on simulated data but less on real data, and it represents the real bottleneck in this field of research. In a larger context, this technique or parts of this technique can be useful for other purposes. The first one is the use of these morphological parameters in the classification of galaxies. The Galaxy Zoo collaboration [36] is trying to classify and label a huge amount of galaxies thanks to volunteers in order to train CNN. These labeled images could be used to train our model to perform automatic classification of galaxies. The use of our method can be interesting to automate this work. From a more personal point of view, this work really allowed me to have a better understanding of the galactic structure and the gravitational lensing phenomenon. I also learned a lot about the image-processing techniques.

Bibliography

[1] I. Newton, *Optiks or a Treatise of the Reflexions, Refractions, Inflexions Colours of Light*. 1704, Book 3, Part 1, Querie 1 (cit. on p. 1).

[2] R. McCormmach, "John michell and henry cavendish: Weighing the stars," vol. 4, no. 2, pp. 126–155, Dec.1968 (cit. on p. 1).

[3] K.-H. Lotze and S. Simionato, "Henry Cavendish and the effect of gravity on propagation of light: A postscript," *The European Physical Journal H*, vol. 46, no. 1, p. 24, Sep. 2021 (cit. on p. 1).

[4] Wikisource. "Translation:on the deflection of a light ray from its rectilinear motion — wikisource." [Online; accessed 7-November-2022]. (2021), [Online]. Available: `https://en.wikisource.org/w/index.php?title=Translation:On_the_Deflection_of_a_Light_Ray_from_its_Rectilinear_Motion&oldid=10821714` (cit. on p. 1).

[5] A. Einstein. "On the influence of gravitation on the propagation of light." [Online; accessed 7-November-2022]. (1911), [Online]. Available: `https://einsteinpapers.press.princeton.edu/vol3-trans/393` (cit. on p. 1).

[6] J.-M. Ginoux, "Albert einstein and the doubling of the deflection of light," *Foundations of Science*, vol. 27, pp. 1–22, Feb. 2021 (cit. on p. 1).

[7] E. S. A. Dyson Frank Watson and D. C., "Ix. a determination of the deflection of light by the sun's gravitational field, from observations made at the total eclipse of may 29, 1919," *Philosophical Transactions of the Royal Society of London.*, vol. Series A, Containing Papers of a Mathematical or Physical Character, no. 220, pp. 291–333, 1920 (cit. on p. 1).

[8] F. Zwicky, "Nebulae as gravitational lenses," *Phys. Rev.*, vol. 51, pp. 290–290, 4 Feb. 1937 (cit. on pp. 1, 2).

[9] R. C. C. Dennis Walsh and R. Weymann, "0957 + 561 a, b: Twin quasistellar objects or gravitational lens?" *Nature*, vol. 279, pp. 381–384, 1979 (cit. on p. 1).

[10] T. York, N. Jackson, I. W. A. Browne, *et al.*, "CLASS B0631+519: last of the Cosmic Lens All-Sky Survey lenses,", vol. 361, no. 1, pp. 259–271, Jul. 2005. arXiv: `astro-ph/0505093 [astro-ph]` (cit. on p. 2).

[11] Y. Tsapras, "Microlensing searches for exoplanets," *Geosciences*, vol. 8, no. 10, p. 365, Sep. 2018 (cit. on p. 2).

[12] R. Massey, T. Kitching, and J. Richard, "The dark matter of gravitational lensing," *Reports on Progress in Physics*, vol. 73, no. 8, p. 086 901, Jul. 2010 (cit. on p. 2).

[13]R. S. Ellis, "Gravitational lensing: A unique probe of dark matter and dark energy," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 368, no. 1914, pp. 967–987, 2010. eprint: https://royalsocietypublishing.org/doi/pdf/10.1098/rsta.2009.0209 (cit. on p. 2).

[14]W. J., "Gravitational lensing in astronomy.," *Living reviews in relativity,*, vol. 1, 1998 (cit. on p. 2).

[15]T. E. Collett, "The population of galaxy–galaxy strong lenses in forthcoming optical imaging surveys," *The Astrophysical Journal*, vol. 811, no. 1, p. 20, Sep. 2015 (cit. on p. 3).

[16]E. Zaborowski, A. Drlica-Wagner, F. Ashmead, *et al.*, *Identification of galaxy-galaxy strong lens candidates in the decam local volume exploration survey using machine learning*, 2022. arXiv: 2210.10802 [astro-ph.GA] (cit. on pp. 3, 11).

[17]J.-F. Claeskens and J. Surdej, "Gravitational lensing in quasar samples," *The Astronomy and Astrophysics Review*, vol. 10, no. 4, pp. 263–311, Mar. 2002 (cit. on p. 3).

[18]P. Magain. "Extragalactic distances." [Online; accessed 24-May-2023]. (2023), [Online]. Available: http://www.astro.ulg.ac.be/cours/magain/AstrophysiqueExtragal/Extragal01E.ppt (cit. on pp. 3, 9).

[19]L. S. Sparke and J. S. Gallagher III, *Galaxies in the Universe: An Introduction*, 2nd ed. Cambridge University Press, 2007 (cit. on p. 10).

[20]S. T. James Binney, *Galactic Dynamics* (Princeton Series in Astrophysics), 2nd. Princeton University Press, 2008 (cit. on p. 10).

[21]L. Alzubaidi, J. Zhang, A. J. Humaidi, *et al.*, "Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions," *Journal of Big Data*, vol. 8, no. 1, p. 53, Mar. 2021 (cit. on p. 11).

[22]M. Saranya, N. Archana, J. Reshma, S. Sangeetha, and M. Varalakshmi, "Object detection and lane changing for self driving car using cnn," in *2022 International Conference on Communication, Computing and Internet of Things (IC3IoT)*, 2022, pp. 1–7 (cit. on p. 11).

[23]B. Kayalibay, G. Jensen, and P. van der Smagt, *Cnn-based segmentation of medical imaging data*, 2017. arXiv: 1701.03056 [cs.CV] (cit. on p. 11).

[24]M. F. Aslan, K. Sabanci, A. Durdu, and M. F. Unlersen, "COVID-19 diagnosis using state-of-the-art CNN architecture features and Bayesian Optimization," *Computers in Biology and Medicine*, vol. 142, p. 105 244, 2022 (cit. on p. 11).

[25]S. Madireddy, N. Ramachandra, N. Li, *et al.*, *A modular deep learning pipeline for galaxy-scale strong gravitational lens detection and modeling*, 2022. arXiv: 1911.03867 [astro-ph.IM] (cit. on p. 11).

[26]S. Rezaei, J. P. McKean, M. Biehl, W. de Roo, and A. Lafontaine, "A machine learning based approach to gravitational lens identification with the International LOFAR Telescope," *Monthly Notices of the Royal Astronomical Society*, vol. 517, no. 1, pp. 1156–1170, Nov. 2022 (cit. on p. 11).

[27]J. Pearson, C. Pennock, and T. Robinson, "Auto-detection of strong gravitational lenses using convolutional neural networks," en, *Emergent Scientist*, vol. 2, p. 1, 2018, Publisher: EDP Sciences (cit. on p. 11).

[28]E. Savary, K. Rojas, M. Maus, *et al.*, "A search for galaxy-scale strong gravitational lenses in the Ultraviolet Near Infrared Optical Northern Survey (UNIONS)," Tech. Rep., Oct. 2021, Publication Title: arXiv e-prints ADS Bibcode: 2021arXiv211011972S Type: article (cit. on pp. 11, 15, 16, 18, 54).

[29]C. Jacobs, T. Collett, K. Glazebrook, *et al.*, "An extended catalog of galaxy–galaxy strong gravitational lenses discovered in des using convolutional neural networks," *The Astrophysical Journal Supplement Series*, vol. 243, no. 1, p. 17, Jul. 2019 (cit. on p. 11).

[30]J. L. Sérsic, "Influence of the atmospheric and instrumental dispersion on the brightness distribution in a galaxy," *Boletin de la Asociacion Argentina de Astronomia La Plata Argentina*, vol. 6, pp. 41–43, Feb. 1963 (cit. on p. 11).

[31]J. M. Lotz, J. Primack, and P. Madau, "A New Nonparametric Approach to Galaxy Morphological Classification,", vol. 128, no. 1, pp. 163–182, Jul. 2004. arXiv: astro-ph/0311352 [astro-ph] (cit. on pp. 11, 33–37).

[32]M. A. Peth, J. M. Lotz, P. E. Freeman, *et al.*, "Beyond spheroids and discs: classifications of CANDELS galaxy structure at $1.4 < z < 2$ via principal component analysis,", vol. 458, no. 1, pp. 963–987, May 2016. arXiv: 1504.01751 [astro-ph.GA] (cit. on pp. 11, 34–36).

[33]P. E. Freeman, R. Izbicki, A. B. Lee, *et al.*, "New image statistics for detecting disturbed galaxy morphologies at high redshift,", vol. 434, no. 1, pp. 282–295, Sep. 2013. arXiv: 1306.1238 [astro-ph.CO] (cit. on p. 11).

[34]A. W. Graham and S. P. Driver, "A concise reference to (projected) sérsic r1/n quantities, including concentration, profile slopes, petrosian indices, and kron magnitudes," *Publications of the Astronomical Society of Australia*, vol. 22, no. 2, pp. 118–127, 2005 (cit. on p. 12).

[35]C. Laigle, H. J. McCracken, O. Ilbert, *et al.*, "THE COSMOS2015 CATALOG: EXPLORING THE 1 &lt iz/i &lt 6 UNIVERSE WITH HALF a MILLION GALAXIES," *The Astrophysical Journal Supplement Series*, vol. 224, no. 2, p. 24, Jun. 2016 (cit. on p. 16).

[36]R. E. Hart, S. P. Bamford, K. W. Willett, *et al.*, "VizieR Online Data Catalog: Galaxy Zoo 2: new classification (Hart+, 2016)," *VizieR Online Data Catalog*, J/MNRAS/461/3663, J/MNRAS/461/3663, Nov. 2017 (cit. on pp. 16, 57).

[37]G. P. Smith, A. Robertson, G. Mahler, *et al.*, "Discovering gravitationally lensed gravitational waves: predicted rates, candidate selection, and localization with the Vera Rubin Observatory," *Monthly Notices of the Royal Astronomical Society*, vol. 520, no. 1, pp. 702–721, Jan. 2023. eprint: https://academic.oup.com/mnras/article-pdf/520/1/702/49032005/stad140.pdf (cit. on p. 18).

[38]L. Bradley, B. Sipőcz, T. Robitaille, *et al.*, *Astropy/photutils: 1.5.0*, version 1.5.0, Jul. 2022 (cit. on p. 27).

[39] Astropy Collaboration, A. M. Price-Whelan, P. L. Lim, *et al.*, "The Astropy Project: Sustaining and Growing a Community-oriented Open-source Project and the Latest Major Release (v5.0) of the Core Package," *apj*, vol. 935, no. 2, 167, p. 167, Aug. 2022. arXiv: 2206.14220 [astro-ph.IM] (cit. on p. 29).

[40] V. Rodriguez-Gomez, G. F. Snyder, J. M. Lotz, *et al.*, "The optical morphologies of galaxies in the IllustrisTNG simulation: a comparison to Pan-STARRS observations,", vol. 483, no. 3, pp. 4140–4159, Mar. 2019. arXiv: 1809.08239 [astro-ph.GA] (cit. on p. 32).

[41] F. Pedregosa, G. Varoquaux, A. Gramfort, *et al.*, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011 (cit. on p. 37).

[42] W. S. Noble, "What is a support vector machine?" *Nature Biotechnology*, vol. 24, no. 12, pp. 1565–1567, Dec. 2006 (cit. on p. 38).

[43] C.-A. Azencott, *Introduction au Machine Learning-2e éd.* Dunod, 2022 (cit. on pp. 38, 39).

[44] G. Biau and E. Scornet, "A random forest guided tour," *Test*, vol. 25, pp. 197–227, 2016 (cit. on p. 38).

[45] W. H. Delashmit, M. T. Manry, *et al.*, "Recent developments in multilayer perceptron neural networks," in *Proceedings of the seventh Annual Memphis Area Engineering and Science Conference, MAESC*, 2005 (cit. on p. 39).

[46] Hunziker, S., Quanz, S. P., Amara, A., and Meyer, M. R., "PCA-based approach for subtracting thermal background emission in high-contrast imaging data," *A&A*, vol. 611, A23, 2018 (cit. on p. 55).

[47] I. T. Jolliffe and J. Cadima, "Principal component analysis: A review and recent developments," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 374, no. 2065, p. 20 150 202, Apr. 2016, Publisher: Royal Society (cit. on p. 55).

List of Figures

List of Tables

List of Listings

Appendix

A

The following images were made thanks to optimal object detection. They are presented here to complete the perception of the multidimensional space parameter presented in section 4.1. Images displayed below are selected among the best separated distributions (lensed and non-lensed)

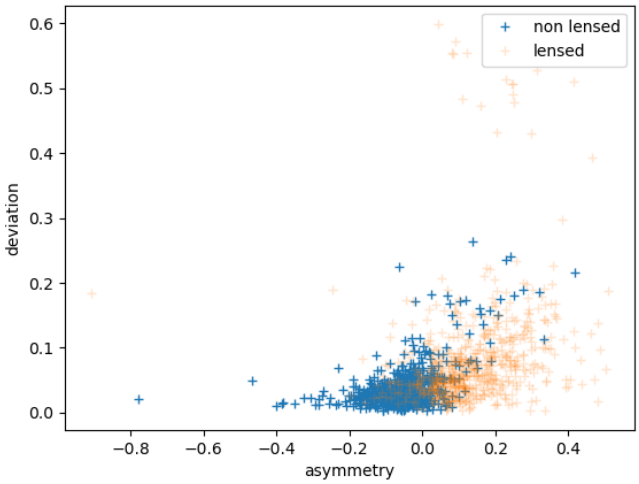

Fig. A.1.: Representation of deviation and asymmetry in the parameter space

These two images represent the projection of the parameters in the new parameter space provided by a PCA as discussed in section 5.2. This method can be use in order to improve the performances of the method described in this work.

Here the 2 images displayed below, are the resulting images of the statmorph computation. The first one is a lensed galaxy while the second one is the same but unlensed galaxy. There are displayed in order to illustrate a new method that can be used to improve the results of this work.

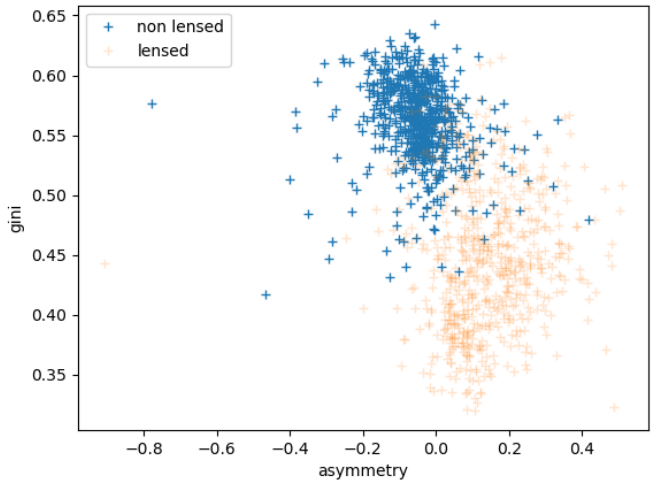

Fig. A.2.: Representation of gini and asymmetry in the parameter space

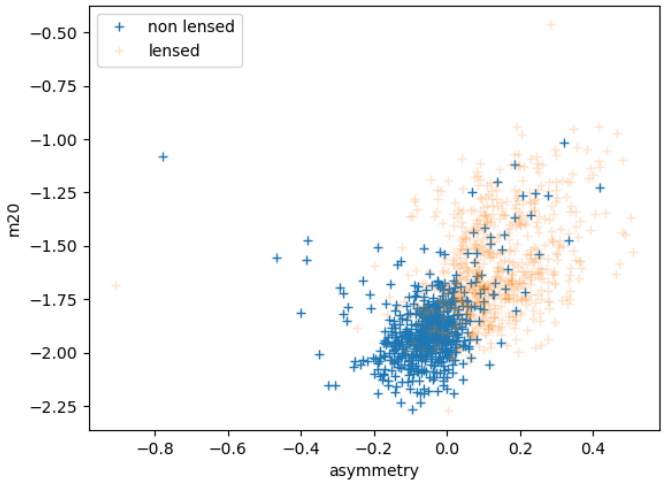

Fig. A.3.: Representation of m20 and asymmetry in the parameter space

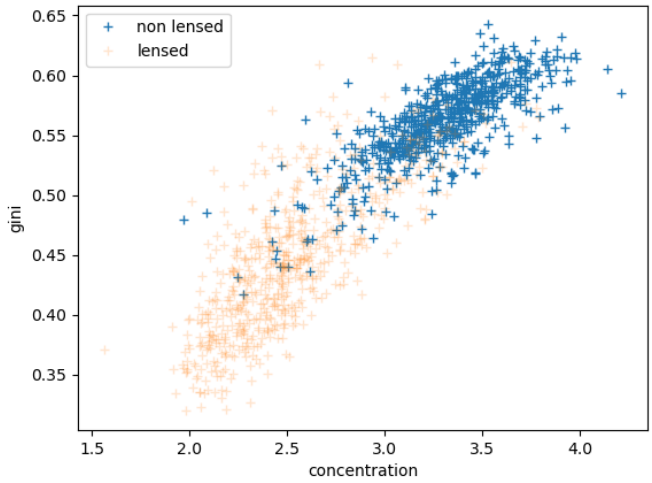

Fig. A.4.: Representation of concentration and gini in the parameter space

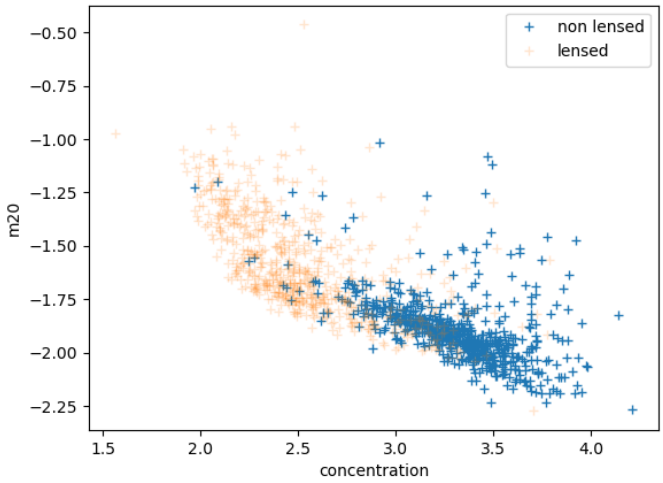

Fig. A.5.: Representation of concentration and m20 in the parameter space

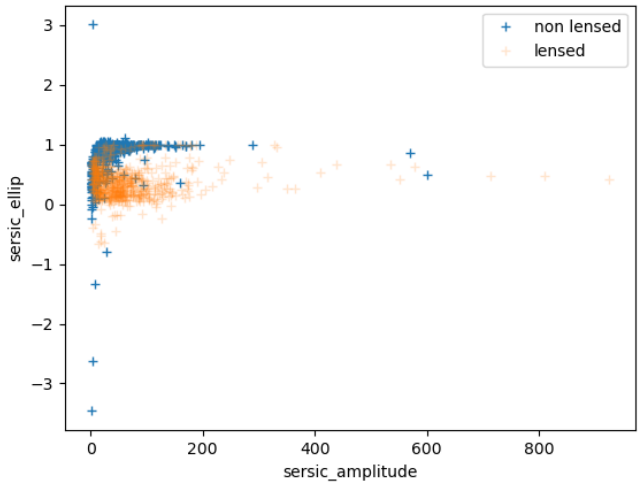

Fig. A.6.: Representation of Sersic amplitude and ellipticity in the parameter space

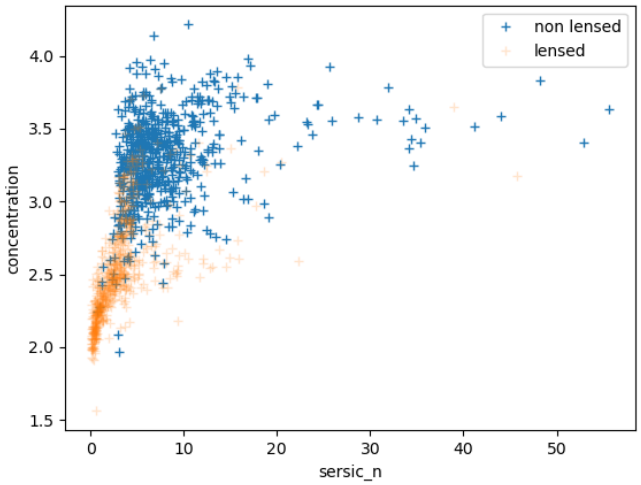

Fig. A.7.: Representation of Sersic index and concentration in the parameter space

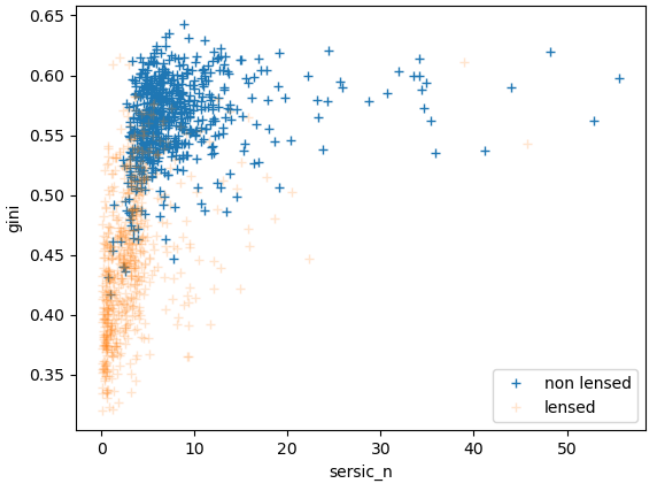

Fig. A.8.: Representation of Sersic index and gini in the parameter space

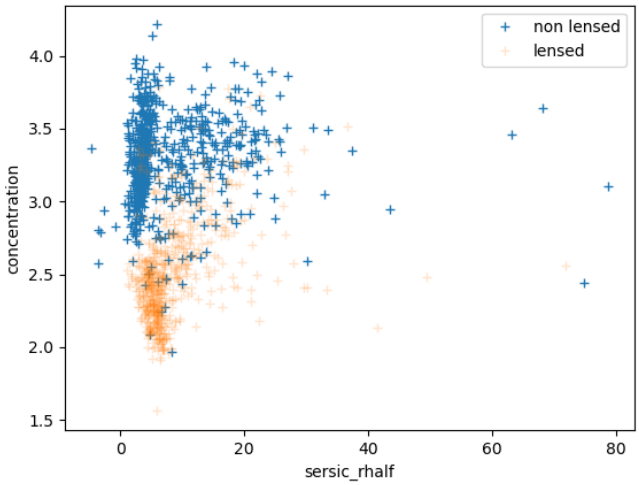

Fig. A.9.: Representation of effective radius and concentration in the parameter space

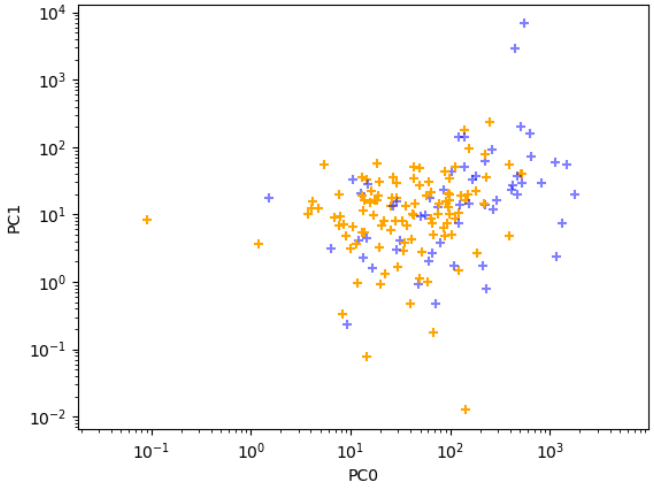

Fig. A.10.: Representation of the 2 first principal components of a PCA on all the parameters used during this work. Blue points stand for lensed galaxies and orange for non-lensed galaxies.

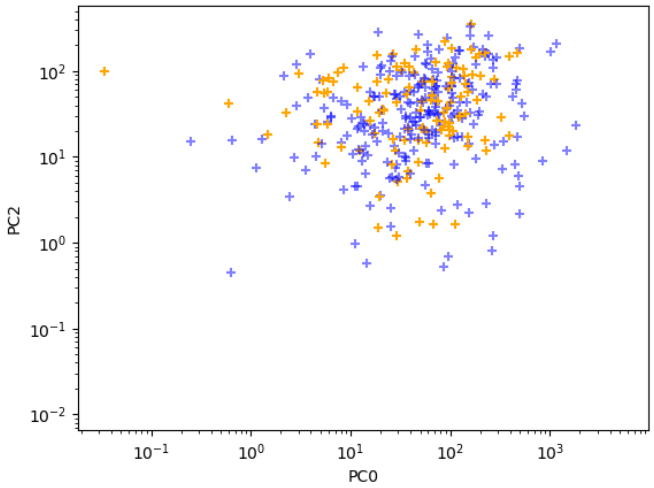

Fig. A.11.: Representation of the first and the third principal components of a PCA on all the parameters used during this work. Blue points stand for lensed galaxies and orange for non-lensed galaxies.

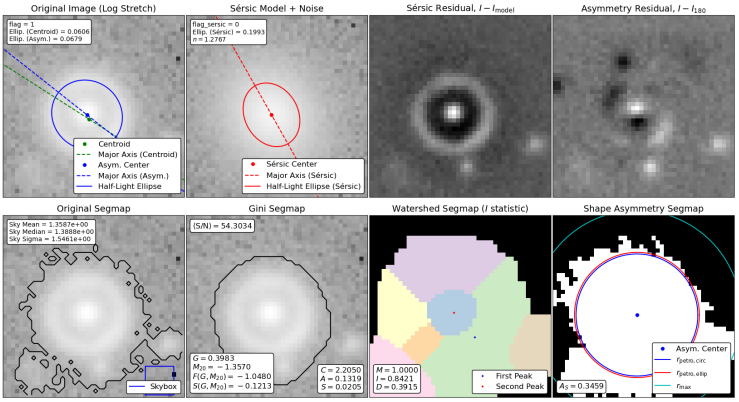

Fig. A.12.: Summary of the `statmorph` outputs with a residual image that need to be compared to the residual image in A.13. It is important to look at the Sersic residual frame since it provides the result of the substraction of the fitted Sersic model from the original image. This image is a lensed galaxy.

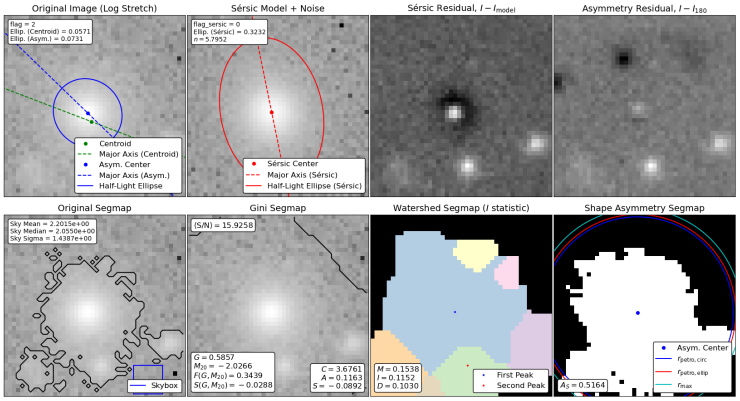

Fig. A.13.: Summary of the `statmorph` outputs with a residual image that need to be compared to the residual image in A.12. It is important to look at the Sersic residual frame since it provides the result of the substraction of the fitted Sersic model from the original image. This image represent the same but unlensed galaxy as the fig A.12.

Colophon

This thesis was typeset with LaTeX 2_ε. It uses the *Clean Thesis* style developed by Ricardo Langner. The design of the *Clean Thesis* style is inspired by user guide documents from Apple Inc.

Download the *Clean Thesis* style at `http://cleanthesis.der-ric.de/`.

Declaration

You can put your declaration here, to declare that you have completed your work solely and only with the help of the references you mentioned.

, June 2023

Laisney Clément